

Comparison of dimensionally-split and multi-dimensional atmospheric transport schemes for long time-steps

Article

Published Version

Creative Commons: Attribution 4.0 (CC-BY)

Open Access

Chen, Y., Weller, H., Pring, S. and Shaw, J. (2017)
Comparison of dimensionally-split and multi-dimensional
atmospheric transport schemes for long time-steps. Quarterly
Journal of the Royal Meteorological Society, 143 (708). pp.
2764-2779. ISSN 1477-870X doi:
<https://doi.org/10.1002/qj.3125> Available at
<https://centaur.reading.ac.uk/71485/>

It is advisable to refer to the publisher's version if you intend to cite from the work. See [Guidance on citing](#).

Published version at: <http://onlinelibrary.wiley.com/doi/10.1002/qj.3125/full>

To link to this article DOI: <http://dx.doi.org/10.1002/qj.3125>

Publisher: Royal Meteorological Society

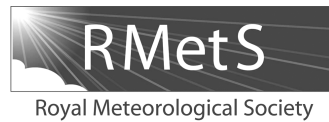
All outputs in CentAUR are protected by Intellectual Property Rights law, including copyright law. Copyright and IPR is retained by the creators or other copyright holders. Terms and conditions for use of this material are defined in the [End User Agreement](#).

www.reading.ac.uk/centaur

CentAUR

Central Archive at the University of Reading

Reading's research outputs online



Comparison of dimensionally split and multi-dimensional atmospheric transport schemes for long time steps

Yumeng Chen,^a Hilary Weller,^{b*} Stephen Pring^c and James Shaw^b

^aDepartment of Mathematics, University of Hamburg, Center for Earth System Research and Sustainability (CEN), University of Hamburg, Germany

^bDepartment of Meteorology, University of Reading, UK

^cMet Office, UK

*Correspondence to: H. Weller, Department of Meteorology, University of Reading, PO Box 243, Reading RG6 6BB, UK.
E-mail: h.weller@reading.ac.uk

Dimensionally split advection schemes are attractive for atmospheric modelling due to their efficiency and accuracy in each spatial dimension. Accurate long time steps can be achieved without significant cost using the flux-form semi-Lagrangian technique. The dimensionally split scheme used in this paper is constructed from the one-dimensional Piecewise Parabolic Method and extended to two dimensions using COSMIC splitting. The dimensionally split scheme is compared with a genuinely multi-dimensional, method-of-lines scheme which, with implicit time-stepping, is stable for Courant numbers significantly larger than 1.

Two-dimensional advection test cases on Cartesian planes are proposed which avoid the complexities of a spherical domain or multi-panel meshes. These are solid-body rotation, horizontal advection over orography and deformational flow. The test cases use distorted non-orthogonal meshes either to represent sloping terrain or to mimic the distortions near cubed-sphere edges.

Mesh distortions are expected to accentuate the errors associated with dimension splitting, however the accuracy of the dimensionally split scheme decreases only a little in the presence of mesh distortions. The dimensionally split scheme also loses some accuracy when long time steps are used. The multi-dimensional scheme is almost entirely insensitive to mesh distortions and asymptotes to second-order accuracy at high resolution. As is expected for implicit time-stepping, phase errors occur when using long time steps but the spatially well-resolved features are advected at the correct speed and the multi-dimensional scheme is always stable.

A naive estimate of computational cost (number of multiplies) reveals that the implicit scheme is the most expensive, particularly for large Courant numbers. If the multi-dimensional scheme is used instead with explicit time-stepping, the Courant number is restricted to less than 1, the accuracy is maintained, and the cost becomes similar to the dimensionally split scheme.

Key Words: multi-dimensional; advection; stability; accuracy; long time steps

Received 11 February 2017; Revised 07 June 2017; Accepted 12 July 2017; Published online in Wiley Online Library

1. Introduction

Many traditional weather and climate models use latitude–longitude meshes, but new models are being developed on quasi-uniform meshes in order to better exploit modern computers (e.g. Skamarock and Gassmann, 2011; Staniforth and Thuburn, 2012; Weller *et al.*, 2012; Lauritzen *et al.*, 2014; Katta *et al.*, 2015). Latitude–longitude meshes need measures to enable long time

steps near the poles such as polar filtering, semi-implicit and semi-Lagrangian methods (Davies *et al.*, 2005). These methods necessarily have large domains of dependence near the poles which lead to poor parallel scaling at high resolution. Quasi-uniform meshes avoid parallel scaling bottlenecks near the poles but lead to non-orthogonal meshes (e.g. the cubed-sphere) or different structures (e.g. icosahedra). Therefore accurate and efficient transport (or advection) schemes on non-orthogonal

meshes are required. There is an abundance of desirable properties of advection schemes, including:

1. Inherent local conservation of the advected quantity.
2. Stability in the presence of large Courant numbers.
3. Accuracy in the presence of large Courant numbers.
4. High-order accuracy.
5. Low computational cost, good parallel scaling and multi-tracer efficiency.
6. Low phase and dispersion errors (advection of all wavenumbers of the advected quantity at close to the correct speed).
7. Low diffusion errors (maintaining amplitude of all wavenumbers of the advected quantity).
8. Boundedness, monotonicity, positivity and maintaining correlations between multiple advected tracers.

Four (important) properties are listed together in the final item because they will not be addressed here. In this paper, we address the issue of conservative, accurate, efficient advection schemes on logically rectangular, non-orthogonal meshes which are stable in the presence of large Courant numbers. These schemes would be particularly relevant for cubed-sphere meshes and for terrain-following meshes. Another novel aspect of this paper is that, for simplicity, we test advection schemes entirely on planar meshes rather than on the sphere, proposing test cases to challenge advection schemes on non-orthogonal meshes without the need to implement meshes in spherical geometry. We assume that a large fraction of the numerical errors associated with terrain-following co-ordinates and the cubed-sphere grid are due to the mesh non-orthogonality, distortion and discontinuity. Other sources of numerical errors, such as those associated with the representation of spherical geometry, are not addressed here.

Dimensionally split schemes (operating separately in each spatial dimension) are attractive for atmospheric modelling due to their efficiency and high accuracy in each spatial dimension (e.g. Leonard *et al.*, 1996; Lin and Rood, 1996; Brassington and Sanderson, 1999; Putman and Lin, 2007; Katta *et al.*, 2015). Inherent conservation is guaranteed by using the flux-form semi-Lagrangian (FFSL or forward in time) technique (e.g. Colella and Woodward, 1984) which integrates the dependent variable over a swept distance upstream of every face in order to calculate fluxes in and out of cells. Accurate long time steps can be achieved without significant cost by calculating cumulative mass fluxes along lines of cells (Leonard *et al.*, 1995). One-dimensional schemes can be used with operator splitting to create dimensionally split, temporally second-order-accurate schemes (e.g. Leonard *et al.*, 1996) on logically rectangular, multidimensional meshes. Dimensionally split schemes have been found to give good accuracy on non-orthogonal meshes such as the cubed-sphere if special treatment is applied over cube edges to improve accuracy: for example, Putman and Lin (2007) use the average of two one-sided schemes at cube edges. Katta *et al.* (2015) use a multi-dimensional transport scheme with dimensionally split reconstructions. They create ghost cells outside each cube panel boundary and achieve accuracy between second and fourth order. Guo *et al.* (2014) report error growth around cube edges using Strang splitting.

Meshes also become non-orthogonal when orography is represented with terrain-following layers. The problem is commonly alleviated by smoothing terrain-following layers to reduce non-orthogonality away from the ground (e.g. Schär *et al.*, 2002), or by using floating Lagrangian vertical coordinates (Lin, 2004). The special treatment applied to dimensionally split schemes for cubed-spheres cannot be applied for terrain-following layers because sloping orography appears throughout the domain. Dimension splitting may account for some of the errors over orography reported by Kent *et al.* (2014), although they do not cite this as a reason for errors.

Dimension-splitting errors on distorted meshes can be eliminated by using genuinely multi-dimensional advection

schemes. These can be either FFSL (i.e. swept area; e.g. Lashley, 2002; Lipscomb and Ringler, 2005; Miura, 2007; Thuburn *et al.*, 2014), method of lines (MOL, discretising space and time separately; e.g. Weller *et al.*, 2009; Skamarock and Gassmann, 2011; Katta *et al.*, 2015; Shaw *et al.*, 2017) or conservative semi-Lagrangian (with conservative re-mapping; e.g. Iske and Kaser, 2004; Zerroukat *et al.*, 2004; Lauritzen *et al.*, 2010). Few FFSL and MOL multi-dimensional schemes have been extended to work with Courant numbers significantly larger than 1, a notable exception being Ullrich and Norman (2014), whose FFSL method is stable for Courant numbers up to around 2.5. FFSL multi-dimensional schemes could be extended to handle larger time steps by integrating the upstream swept volume over a large upstream volume, interacting with a large number of upstream cells. However, the integration cost would be proportional to the time step since an upstream swept volume would overlap with more cells. MOL multi-dimensional schemes can be extended to work with Courant numbers larger than 1 by using implicit time-stepping. This will increase the computational cost per tracer advected since the solution of a matrix equation would be needed for every advected tracer. Other disadvantages of implicit time-stepping are the large phase errors when long time steps are used (e.g. Durran and Blossey, 2012; Lock *et al.*, 2014), the difficulty of achieving monotonicity, and the parallelization cost of linear solvers. This is in contrast to semi-Lagrangian or FFSL schemes which maintain accuracy with long time steps (Purnell, 1976; Pudykiewicz and Staniforth, 1984; Leonard *et al.*, 1995) although monotonicity with long time steps is still challenging (Bott, 2010). Conservative semi-Lagrangian (e.g. Zerroukat *et al.*, 2004) naturally extends to long time steps but the conservative remapping is complicated and expensive, particularly on non-rectangular meshes, and will not be investigated here. Lauritzen *et al.* (2014) described how the FFSL technique with a long time step can be made equivalent to the conservative semi-Lagrangian.

It is therefore not clear what approach should be taken for achieving long time steps when advecting multiple tracers on distorted meshes. In this paper, we show the effect of dimension-splitting errors using a FFSL dimensionally split scheme on a number of test cases which use distorted and undistorted meshes, and compare with a genuinely multi-dimensional implicit MOL scheme using large and small Courant numbers.

The theoretical properties of dimensionally split advection schemes are often tested on uniform, orthogonal meshes (e.g. Leonard *et al.*, 1996) which is inadequate for a scheme that will eventually be used on a cubed-sphere. Developing a transport scheme to the extent that it can be used on a multi-panel cubed-sphere with special treatment of cube edges is a considerable undertaking. Hence there is a need for more challenging advection test cases which are simpler to implement, without the need for spherical meshes. We therefore propose some modifications of existing test cases to use distorted meshes, or distorted co-ordinate systems, on a logically rectangular, two-dimensional plane. These test cases will mimic the problems encountered at cubed-sphere edges.

The long-time-step-permitting, dimensionally split scheme and the long-time-step-permitting multi-dimensional scheme are defined in section 2. In section 3 we present results of three advection test cases on distorted meshes in two-dimensional planes using Courant numbers above and below 1. These are: the solid-body rotation test case of Leonard *et al.* (1996), modified to use a mesh (or co-ordinate system) with distortions similar to a cubed-sphere (section 3.1); the horizontal advection test case over orography (Schär *et al.*, 2002), examining sensitivity to time step, resolution and mountain height, all on the basic terrain following mesh without smoothing of terrain following layers (section 3.2); and a modification of the deformational flow test case of Lauritzen *et al.* (2012) for a periodic rectangular plane (section 3.3). Some estimates are made of computational cost in section 3.4 and final conclusions are drawn in section 4.

2. Transport schemes

We present two conservative advection schemes suitable for long time steps (stable for Courant numbers significantly larger than 1) for solving the linear advection equation:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\mathbf{u}\phi) = 0, \quad (1)$$

where the dependent variable ϕ is advected by a divergence free velocity field $\mathbf{u}(\mathbf{x}, t)$. The dimensionally split scheme is the piecewise parabolic method (PPM; Colella and Woodward, 1984) which uses the FFSL approach extended to long time steps following Leonard *et al.* (1995) with COSMIC splitting Leonard *et al.* (1996) to extend PPM to two dimensions. The multi-dimensional scheme uses the method-of-lines approach (treating space and time independently). The second-order accurate spatial discretisation of Shaw *et al.* (2017) is combined with an explicit, second-order Runge–Kutta time-stepping scheme and with an implicit, Crank–Nicholson time-stepping scheme to allow long time steps. Neither scheme has monotonicity or positivity preservation. The dimensionally split scheme is third-order accurate in one dimension but the COSMIC splitting reduces the temporal order to two. The multi-dimensional scheme uses cubic interpolations but uses cell centroids as Gauss points so the order of accuracy is limited to two. When explicit time-stepping is used, the cost of both schemes is similar (section 3.4). These schemes are therefore considered suitable for comparison.

The code for dimensionally splitting scheme is available at <https://github.com/yumengch/COSMIC-splitting>. The multi-dimensional scheme is implemented using OpenFOAM (2016) and is available at <https://github.com/AtmosFOAM/>. The set-up of the test cases is available at <https://github.com/hilaryweller/0/multiDadvectCases> (all accessed 26 July 2017).

2.1. One-dimensional PPM with long time steps

We describe a long-time-step version of PPM for solving the one-dimensional advection equation:

$$\frac{\partial \phi}{\partial t} + \frac{\partial u\phi}{\partial x} = 0. \quad (2)$$

Colella and Woodward (1984) defined PPM with monotonicity constraints and for variable resolution but for simplicity (and for comparison with the multi-dimensional scheme) we will define PPM without monotonicity constraints and for a fixed resolution Δx , and time step Δt . With these restrictions, PPM should be fourth-order accurate in one dimension for vanishing time step. We define the dependent variable, $\phi_i^{(n)}$, to be the mean value of ϕ in cell i at time-level n where $x_i = i\Delta x$ and $t = n\Delta t$. Since PPM is a flux-form finite-volume method, $\phi_i^{(n)}$ is updated using:

$$\phi_i^{(n+1)} = \phi_i^{(n)} + X_C(\phi) \quad (3)$$

$$= \phi_i^{(n)} - \frac{u_{i+1/2}\phi_{i+1/2} - u_{i-1/2}\phi_{i-1/2}}{\Delta x}, \quad (4)$$

where $X_C(\phi)$ is the conservative advection operator for ϕ in the x direction. The fluxes, $u_{i\pm 1/2}\phi_{i\pm 1/2}$, are found by integrating a piecewise polynomial p , along the distance travelled in each time step upwind of cell boundary $x_{i\pm 1/2}$. The polynomial is defined in each cell i , such that:

$$\phi_i = \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} p_i(x) dx \quad (5)$$

by

$$p_i(x) = p_{i-1/2} + \xi \left[p_{i+1/2} - p_{i-1/2} + (1-\xi)6 \times \left\{ \phi_i - \frac{1}{2}(p_{i-1/2} + p_{i+1/2}) \right\} \right], \quad (6)$$

where $\xi = (x - x_{i-1/2})/\Delta x$ and

$$p_{i+1/2} = \frac{7}{12}(\phi_i + \phi_{i+1}) - \frac{1}{12}(\phi_{i+2} + \phi_{i-1}). \quad (7)$$

In order to cope with long time steps, we follow Leonard *et al.* (1995) and divide the Courant number into a signed integer part, c_N , and a remainder, c_r . The departure point of location $x_{i-1/2}$ is thus computed as $x_d = x_{i-1/2} - u_{i-1/2}\Delta t$, and the departure cell, for cell edge $i - \frac{1}{2}$, is $i_d = i - c_N - 1$ for $u_{i-1/2} > 0$ and $i_d = i - c_N$ for $u_{i-1/2} < 0$. Then for $u_{i-1/2} > 0$, the flux through $x_{i-1/2}$ between times $n\Delta t$ and $(n+1)\Delta t$ is:

$$u_{i-1/2}\phi_{i-1/2} = \frac{1}{\Delta t} \int_{x_d}^{x_{i-1/2}} p(x) dx = \frac{1}{\Delta t} \left(M_{i-1/2} - M_{i-c_N-1/2} + \int_{x_d}^{x_{i_d+1/2}} p(x) dx \right), \quad (8)$$

where $M_{i-1/2}$ is the cumulative mass from the start point to position $x_{i-1/2}$:

$$M_{i-1/2} = \sum_{k < i} \Delta x \phi_k. \quad (9)$$

This departure point calculation assumes that the velocity is uniform on the computational mesh which has a first-order error which could be particularly damaging for long Courant numbers, when the wrong departure cell could be found. The departure point calculation could be improved by using a second-order predictor–corrector departure point calculation (e.g. Melvin *et al.*, 2010).

The velocity is derived from a stream function and the Jacobian of the co-ordinate transform:

$$\begin{pmatrix} u \\ v \end{pmatrix} = J \begin{pmatrix} \Psi_y \\ -\Psi_x \end{pmatrix}. \quad (10)$$

For stability, the time step is restricted by the deformational Courant number:

$$c_d = \Delta t \max \left(\left| \frac{\partial u}{\partial x} \right|, \left| \frac{\partial u}{\partial y} \right|, \left| \frac{\partial u}{\partial z} \right|, \left| \frac{\partial v}{\partial x} \right|, \left| \frac{\partial v}{\partial y} \right|, \left| \frac{\partial v}{\partial z} \right|, \left| \frac{\partial w}{\partial x} \right|, \left| \frac{\partial w}{\partial y} \right|, \left| \frac{\partial w}{\partial z} \right| \right) \quad (11)$$

(Pudykiewicz and Staniforth, 1984) such that $c_d \leq 1$.

2.2. COSMIC splitting

COSMIC operator splitting (Leonard *et al.*, 1996) allows single-stage, one-dimensional schemes such as PPM to be used stably in two or more dimensions whilst retaining conservation, constancy preservation and second-order accuracy (on orthogonal meshes). As we are now considering two spatial dimensions, we define ϕ_{ij} , u_{ij} and v_{ij} , the values of ϕ and the velocity components, u and v in cell (i, j) where $x = i\Delta x$ and $y = j\Delta y$. COSMIC splitting uses both advective (A) and conservative (C) advection operators in the x and y directions:

$$X_C(\phi) = -\frac{1}{\Delta x} (u_e \phi_e - u_w \phi_w), \quad (12)$$

$$Y_C(\phi) = -\frac{1}{\Delta y} (v_n \phi_n - v_s \phi_s), \quad (13)$$

$$X_A(\phi) = X_C(\phi) + \frac{\phi_{ij}}{\Delta x} (u_e - u_w), \quad (14)$$

$$Y_A(\phi) = Y_C(\phi) + \frac{\phi_{ij}}{\Delta y} (v_n - v_s), \quad (15)$$

where $\phi_n = \phi_{i,j+1/2}$, $\phi_s = \phi_{i,j-1/2}$, $\phi_e = \phi_{i+1/2,j}$, $\phi_w = \phi_{i-1/2,j}$, $v_n = v_{i,j+1/2}$, $v_s = v_{i,j-1/2}$, $u_e = u_{i+1/2,j}$ and $u_w = u_{i-1/2,j}$ are the values of ϕ , u and v at the (n, s, e and w) cell boundaries. If COSMIC is being used to extend PPM to two spatial dimensions, then $\phi_{n,s,e,w}$ are calculated from Eq. (8). Assuming C-grid staggering, v_n , v_s , u_e and u_w are dependent variables. Instead of using cell-centred velocity (Lin and Rood, 1996) or upwind velocity (Leonard *et al.*, 1996), the advective operators are calculated in a similar manner to Lin (2004).

Mesh distortions can be included in the advection equation with a co-ordinate transform with Jacobian J :

$$\frac{\partial |J|^{-1} \phi}{\partial t} + \frac{\partial |J|^{-1} u \phi}{\partial x} + \frac{\partial |J|^{-1} v \phi}{\partial y} = 0. \quad (16)$$

The operators X_C , Y_C , X_A and Y_A are then:

$$X_C(\phi) = -\frac{1}{\Delta x} (|J|_e^{-1} u_e \phi_e - |J|_w^{-1} u_w \phi_w), \quad (17)$$

$$Y_C(\phi) = -\frac{1}{\Delta y} (|J|_n^{-1} v_n \phi_n - |J|_s^{-1} v_s \phi_s), \quad (18)$$

$$X_A(\phi) = X_C(\phi) + \frac{\phi_{ij}}{\Delta x} (|J|_e^{-1} u_e - |J|_w^{-1} u_w), \quad (19)$$

$$Y_A(\phi) = Y_C(\phi) + \frac{\phi_{ij}}{\Delta y} (|J|_n^{-1} v_n - |J|_s^{-1} v_s), \quad (20)$$

which are combined to update ϕ_{ij} in each cell by:

$$\begin{aligned} \phi_{ij}^{(n+1)} = & \phi_{ij}^n + |J|_{ij} X_C \left\{ \phi_{ij}^{(n)} + \frac{|J|_{ij}}{2} Y_A \left(\phi_{ij}^{(n)} \right) \right\} \\ & + |J|_{ij} Y_C \left\{ \phi_{ij}^{(n)} + \frac{|J|_{ij}}{2} X_A \left(\phi_{ij}^{(n)} \right) \right\}. \end{aligned} \quad (21)$$

where $|J|_{ij}$ is the determinant of Jacobian at the centre of cell (i, j) , and Δx and Δy give the cell size in uniform computational domain.

2.3. Multi-dimensional method-of-lines (MOL) scheme

The MOL scheme uses the finite volume method for arbitrary meshes and is implemented in OpenFOAM (2016). This uses a cubic upwind spatial discretization (Weller and Shahrokh, 2014; Shaw *et al.*, 2017) combined with either implicit Crank–Nicholson in time or explicit RK2 (see below). Although the interpolation uses a cubic polynomial, cell centre values are approximated as cell average values and face centre values are approximated as face average values so the method is limited to second-order accuracy in space. The advection scheme uses Gauss's divergence theorem to approximate the divergence term of the advection equation:

$$\nabla \cdot (\mathbf{u} \phi) \approx \frac{1}{V} \sum_{f \in c} \phi_f \mathbf{u}_f \cdot \mathbf{S}_f, \quad (22)$$

where V is the cell volume, the summation is over all faces, f , of cell c , ϕ_f is the value of the dependent variable, ϕ interpolated onto face f , \mathbf{u}_f is the velocity at face f and \mathbf{S}_f is the vector normal to face f with magnitude equal to the area of face f (i.e. the face area vector). The advection scheme uses a fit to a 2D (or 3D) polynomial using an upwind-biased stencil of cells (Figure 1) in order to interpolate from known cell values onto a face. In 2D, the cubic polynomial is:

$$\begin{aligned} \phi = & a_0 + a_1 x + a_2 y + a_3 x^2 + a_4 xy \\ & + a_5 y^2 + a_6 x^3 + a_7 x^2 y + a_8 xy^2, \end{aligned} \quad (23)$$

omitting the y^3 term, where x is the direction normal to a cell face and y is perpendicular to x . (The y^3 term is omitted because

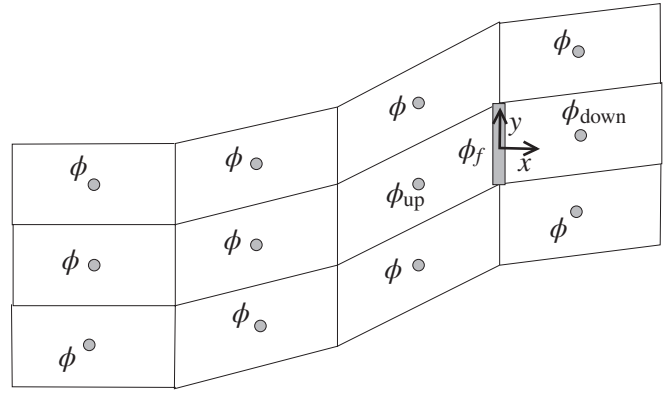


Figure 1. Stencil of upwind-biased cells for interpolating onto face f using a 2D cubic polynomial for a rectangular mesh structure for the multi-dimensional scheme. The tracer, ϕ , is stored at cell centres (grey dots). 'up' and 'down' refer to upwind and downwind cells relative to the central face, f , and x - y is the co-ordinate system relative to f .

it cannot be set with a stencil that is narrow in the direction of the flow.) Coefficients a_i are set from a least-squares fit to the cell data in the stencil. The least-squares problem involves a $9 \times m$ matrix singular value decomposition (where m is the size of the stencil) for every face and for both orientations of each face. However this is purely a geometric calculation and is therefore a pre-processing activity since the mesh is fixed. This generates a set of weights for calculating ϕ_f from the cell values in the stencil, leaving m multiplies for each face for each call of the advection operator. The stencils are found for three-dimensional, arbitrarily structured meshes by finding the face(s) closest to upwind of the face we are interpolating onto, taking the two cells either side of the upwind face(s) and then taking the vertex neighbours of those central cells (Figure 1). For each face there are two possible stencils depending on the upwind direction. Both of the stencils are stored and the interpolation weights for both stencils are calculated.

In order to ensure that the fit is accurate in the cells either side of face f and to ensure that the values in these adjacent cells have the strongest control over ϕ_f , rows associated with these values in the least-squares fit matrix are weighted a factor of 1000 relative to the other rows (following Lashley, 2002). This does not affect the order of accuracy. Mathematically, an arbitrarily large value of the weight can be used to ensure that the fit goes exactly through the upwind and downwind cell. However if a value too large is used, the singular value problem becomes ill-conditioned. The stabilization procedures described by Shaw *et al.* (2017) are in the code but are not activated for these test cases because the meshes are sufficiently regular. The value ϕ_f is then calculated as a higher-order correction to first-order upwind:

$$\phi_f = \phi_{up} + \sum_c w_c \phi_c, \quad (24)$$

where w_c are the weights for each cell of the stencil calculated from the least-squares fit, with w_{up} reduced by 1 to make the fit a correction on upwind.

2.3.1. Implicit and explicit time-stepping

Assuming that the velocity field and mesh are constant in time, the explicit Runge–Kutta 2 (Heun) time-stepping is defined as

$$\frac{\phi' - \phi^n}{\Delta t} = -\frac{1}{V} \sum_{f \in c} \left(\phi_{up}^n + \sum_c w_c \phi_c^n \right) \mathbf{u}_f \cdot \mathbf{S}_f, \quad (25)$$

$$\begin{aligned} \frac{\phi^{n+1} - \phi^n}{\Delta t} = & -\frac{1}{2V} \sum_{f \in c} \left(\phi_{up}^n + \phi'_{up} \right. \\ & \left. + \sum_c w_c \phi_c^n + \sum_c w_c \phi_c' \right) \mathbf{u}_f \cdot \mathbf{S}_f. \end{aligned} \quad (26)$$

The trapezoidal implicit or Crank–Nicolson time-stepping leads to a matrix equation which needs to be solved to find all the ϕ s at the next time step. In order to ensure that the matrix is diagonally dominant for arbitrary time steps, the cubic interpolation applied is a deferred correction on first-order upwind so that only the coefficient corresponding to the upwind cells are included in the matrix. This means that more than one implicit solves are needed per time step so that the higher-order terms are solved to be second-order accurate in time. If the Courant number is less than or close to 1, we use two implicit solves per time step. Consequently, assuming that the velocity field and mesh are constant in time, the time-stepping scheme is defined as:

$$\frac{\phi' - \phi^n}{\Delta t} = -\frac{1}{2V} \sum_{f \in c} \left(\phi_{\text{up}}^n + \phi'_{\text{up}} + 2 \sum_c w_c \phi_c^n \right) \mathbf{u}_f \cdot \mathbf{S}_f, \quad (27)$$

$$\begin{aligned} \frac{\phi^{n+1} - \phi^n}{\Delta t} = & -\frac{1}{2V} \sum_{f \in c} \left(\phi_{\text{up}}^n + \phi_{\text{up}}^{n+1} \right. \\ & \left. \times \sum_c w_c \phi_c^n + \sum_c w_c \phi_c^{n+1} \right) \mathbf{u}_f \cdot \mathbf{S}_f. \end{aligned} \quad (28)$$

For Courant numbers larger than 1, we use four implicit solves per time step, although sensitivity to this choice has not been investigated.

2.3.2. Matrix solvers and tolerances

If this implicit scheme is applied on a logically rectangular, two-dimensional mesh with horizontal and vertical Courant numbers c_x and c_z , then the diagonal coefficients of the matrix would be $1 + c_x/2 + c_z/2$ and, assuming two upwind directions, there would be exactly two off-diagonal elements, $-c_x/2$ and $-c_z/2$. Consequently the matrix is very sparse, asymmetric and diagonally dominant for all time steps. It is solved using the OpenFOAM bi-conjugate gradient solver using DILU pre-conditioning to a tolerance of 10^{-8} every iteration. Sensitivity to the solver or solver tolerance have not been investigated. Information about the number of solver iterations for different test cases is given in section 3.4.

3. Results of test cases in planar geometry

In order to make test cases as simple as possible without the need for incorporating spherical geometry, multi-panel meshes or non-rectangular cells, our computational domain consists of a periodic two-dimensional plane with deformations and discontinuities in the co-ordinate system (or mesh) mimicking the kind of distortions which are produced near cubed-sphere edges. We also use a two-dimensional vertical slice test case over orography using terrain-following co-ordinates (or terrain-following meshes).

A range of test cases are undertaken on uniform and distorted meshes using the dimensionally split scheme and the multi-dimensional scheme in order to evaluate the influences of mesh distortions, the validity of using a dimensionally split scheme on a distorted mesh, and the schemes' accuracy and stability for long time steps.

3.1. Solid-body rotation

The solid-body rotation test case of Leonard *et al.* (1996) is used to compare the accuracy of the dimensionally split and multi-dimensional schemes on orthogonal and non-orthogonal meshes. We define this test case on a domain that is $10^4 \times 10^4 \text{ m}^2$. The velocity is defined by numerically differentiating the streamfunction which can be defined at mesh vertices, \mathbf{x} , by:

$$\psi(\mathbf{x}, t) = A|\mathbf{x} - \mathbf{x}_c|^2, \quad (29)$$

where \mathbf{x}_c is the centre of the domain, and $A = 5\pi/3000 \text{ s}^{-1}$ so that the angular velocity is $2A$. The initial tracer takes a Gaussian distribution in order to ensure that all advection schemes achieve their theoretical order of accuracy:

$$\phi(\mathbf{x}) = \exp\left(-\frac{1}{2} \frac{|\mathbf{x} - \mathbf{x}_\phi|^2}{r_\phi^2}\right), \quad (30)$$

where $\mathbf{x}_\phi = \mathbf{x}_c + r_{c\phi}\mathbf{j}$ is the initial centre of the tracer distribution, $r_{c\phi} = 2500 \text{ m}$, $r_\phi = 500 \text{ m}$ and \mathbf{j} is the unit vector in the y direction. The analytic solution has the same tracer distribution, but with the centre of the tracer at

$$\mathbf{x}_\phi = \mathbf{x}_c + r_{c\phi} \begin{pmatrix} \cos(\pi/2 + 2At) \\ \sin(\pi/2 + 2At) \end{pmatrix}, \quad (31)$$

and the Gaussian rotates anti-clockwise exactly one revolution in 600 s.

The solid-body rotation test case is performed on a uniform, orthogonal mesh and on a non-orthogonal mesh on a plane with non-orthogonality similar to that of a cubed-sphere mesh. For the dimensionally split scheme, non-orthogonality is achieved using the co-ordinate transform:

$$X = x, \quad Y = \begin{cases} y_m \left(1 + \frac{y-f}{2y_m-f}\right) & \text{for } y \geq f, \\ y_m \left(1 + \frac{y-f}{f}\right) & \text{for } y < f, \end{cases} \quad (32)$$

where $y_m = 5000 \text{ m}$ and f is the equation for a y position of uniform Y half way up the domain. In order to create angles of 120° in the mesh, similar to a cubed-sphere, f is given by:

$$f = \begin{cases} y_m \left(1 + \frac{1}{2\sqrt{3}}\right) - \frac{x}{\sqrt{3}} & \text{for } x \leq x_m, \\ y_m \left(1 - \frac{1}{2\sqrt{3}}\right) + \frac{x - x_m}{\sqrt{3}} & \text{for } x > x_m, \end{cases} \quad (33)$$

where $x_m = 5000 \text{ m}$. For a 50×50 mesh this gives the x and y co-ordinate locations as shown in Figure 2. The multi-dimensional scheme model uses Cartesian co-ordinates and a distorted mesh rather than a non-orthogonal co-ordinate system on a Cartesian mesh. However, this does not affect the numerical results assuming that the co-ordinate transforms are implemented in a consistent way to the distorted mesh in Cartesian co-ordinates.

For the dimensionally split scheme, bi-periodic boundary conditions are applied. For the multi-dimensional scheme, it was more straightforward to impose fixed-value boundary conditions of $\phi = 0$ where the velocity is into the domain and zero normal gradient where the velocity is out of the domain. However, ϕ remains almost zero near the boundaries so these boundary conditions do not affect the accuracy.

Results of this test case on the orthogonal and non-orthogonal meshes of 100×100 cells with $\Delta t = 1 \text{ s}$ are shown in Figure 3 for both advection schemes (which gives a maximum Courant number close to 1). The contours show the tracer value every 100 s and the colours show the errors summed every 100 s. The dimensionally split scheme outperforms the multi-dimensional scheme on both meshes due to the higher-order accuracy of the split scheme. The dimensionally split scheme introduces a small error at 300 s where the tracer goes through the change in direction of the mesh which would be ameliorated if we were using monotonicity constraints. The second-order, multi-dimensional scheme shows phase lag but errors are almost entirely insensitive to the mesh distortions. The results in Figure 3 use implicit time-stepping but the results using explicit time-stepping are similar (not shown).

The multi-dimensional and dimensionally split schemes take very different approaches to handling large Courant numbers. The

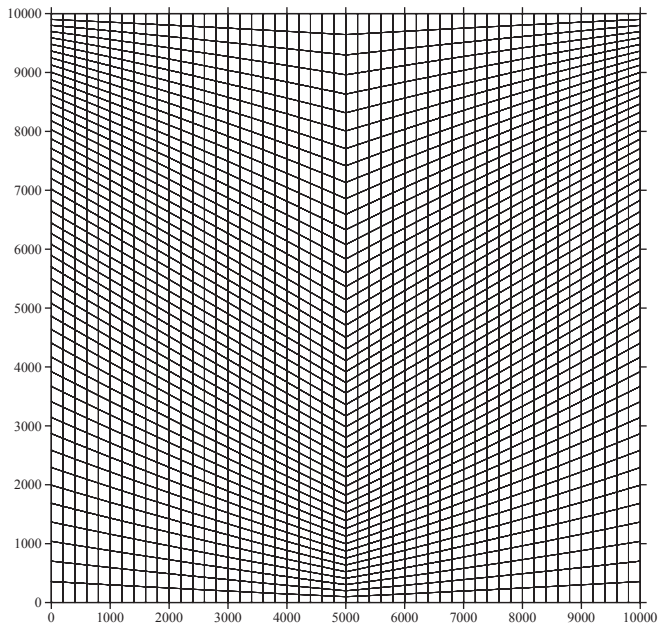


Figure 2. The non-orthogonal mesh on a plane with 50×50 mesh points and angles up to 120° designed to have similar angles to a cubed-sphere mesh.

multi-dimensional scheme uses implicit time-stepping whereas the dimensionally split scheme uses a FFSL approach, integrating over a line of cells in order to calculate the flux across a face. Implicit schemes are known to suffer from phase errors (e.g. Durran and Blossey, 2012; Lock *et al.*, 2014) for long time steps, whereas the accuracy of semi-Lagrangian is less sensitive to time-step (Pudykiewicz and Staniforth, 1984). Therefore we present results of both schemes on orthogonal and non-orthogonal meshes for time steps ten times those used in Figure 3 ($\Delta t = 10$ s) giving maximum Courant numbers of around 10 using 100×100 cells in Figure 4. The error of the multi-dimensional scheme is again much larger than the dimensionally split scheme on both meshes. The dimensionally split scheme is accurate at large Courant numbers despite the first-order calculation of departure points. However, the dimensionally split scheme introduces oscillations on the non-orthogonal mesh, particularly where the mesh changes direction, whereas for the multi-dimensional scheme, errors are not strongly affected by the non-orthogonality. Figure 4 clearly shows phase errors of the implicit time-stepping but, despite the large Courant number, the well-resolved part of the profile is propagating at close to the correct speed. Dispersion analysis (Lock *et al.*, 2014) shows that high-frequency oscillations (which are poorly resolved in time and space) will be slowed dramatically, but fast-moving features which are *well-resolved in space* will propagate at a much more realistic speed, supporting the results shown in Figure 4. The explicit multi-dimensional scheme is not stable for this time step.

In order to compare convergence with resolution of the different numerical methods, we use the ℓ_2 and ℓ_∞ error norms defined in the usual way:

$$\ell_2 = \sqrt{\int_V (\phi - \phi_T)^2 dV} / \sqrt{\int_V \phi_T^2 dV}, \quad (34)$$

$$\ell_\infty = \max |\phi - \phi_T| / \max |\phi_T|, \quad (35)$$

where ϕ_T is the analytic solution and the integrations and maxima are over the whole domain, with volume V . Figure 5 shows convergence with resolution of the ℓ_2 and ℓ_∞ error measures for meshes of 50×50 , 100×100 , 200×200 and 400×400 cells with time steps scaled in order to maintain a maximum Courant number of 1 ($\Delta t = 2, 1, 0.5, 0.25$ s) or scaled to achieve a maximum Courant number of 10 ($\Delta t = 20, 10, 5, 2.5$ s). The error norms are calculated at $t = 500$ s, when the tracer has made $5/6$ of one revolution in order to avoid error cancellation.

On both orthogonal and non-orthogonal meshes, the multi-dimensional scheme has second-order convergence once errors are low enough to avoid error saturation (stable errors are bounded at around 1). With a large Courant number, both schemes are less accurate. For the multi-dimensional scheme, this is due to phase errors of the implicit time-stepping whereas for the multi-dimensional scheme the first-order errors in calculating the departure point and trajectory will be emerging and there could also be significant errors from the second-order COSMIC splitting. On the orthogonal mesh, the dimensionally split scheme has third-order convergence for the Courant number close to 1 and second-order for the larger Courant number (consistent with the results of Colella and Woodward, 1984; Leonard *et al.*, 1996). On the non-orthogonal mesh, the dimensionally split scheme has second-order convergence for both Courant numbers. Results of the explicit multi-dimensional scheme are also shown in Figure 5 for the shorter time step on the non-orthogonal mesh. The explicit RK2 scheme is more accurate than the implicit (Crank–Nicolson) scheme with the same order of convergence, but does not come close to the accuracy of the dimensionally split scheme.

The different time-stepping schemes of the two models also affect convergence with time step. The ℓ_2 and ℓ_∞ error measures as a function of time step for meshes of 100×100 cells are shown in Figure 6. The dimensionally split scheme, which uses FFSL time-stepping, has errors reducing as time step increases, up to a Courant number of 2 ($\Delta t = 2$ s), whereas the multi-dimensional scheme using implicit method-of-lines to treat space and time separately always has error reducing as time steps reduce. The FFSL technique discretizes space and time together and the error is not very sensitive to time step. However, the shorter the time step, more time steps need to be taken, and so errors can actually accumulate more from taking more time steps. (This is consistent with the order of accuracy of semi-Lagrangian being $\Delta x^p / \Delta t$ for interpolation using polynomials of degree p , as described by Durran, 2010). The explicit version of the multi-dimensional scheme has errors that decrease as time step increases due to the spatial discretization errors that are being introduced at every time step.

In summary, the dimensionally split scheme has excellent behaviour at large and small Courant numbers on the orthogonal meshes with up to third-order convergence for small Courant numbers, and the errors increase and order of convergence decreases on non-orthogonal meshes. In contrast, the multi-dimensional scheme is insensitive to the orthogonality, converges with second-order as expected, and suffers from phase errors at large Courant numbers due to the use of Crank–Nicolson implicit time-stepping.

3.2. Horizontal advection over orography

Non-orthogonal meshes (or co-ordinate systems) are usually necessary for representing orography with terrain-following coordinates. This is a challenge for dimensionally split schemes. Horizontal–vertical split schemes are commonly used in this context (e.g. Dennis *et al.*, 2012; Wan *et al.*, 2013) since most weather/climate models use higher resolution in the vertical than in the horizontal. We present results of the Schär *et al.* (2002) horizontal advection over orography test case for a range of resolutions and Courant numbers for the dimensionally split and multi-dimensional schemes.

All simulations use basic terrain-following co-ordinates (BTF; Gal-Chen and Somerville, 1975) in order to present a challenging test case that maximizes the non-orthogonality that can be introduced by terrain-following coordinates. The transformation is given by:

$$X = x, \quad Z = H \frac{z - h(x)}{H - h(x)}, \quad (36)$$

where H is the domain height and h is the terrain height. The test case uses a domain of width 300 km, height, $H = 25$ km and a

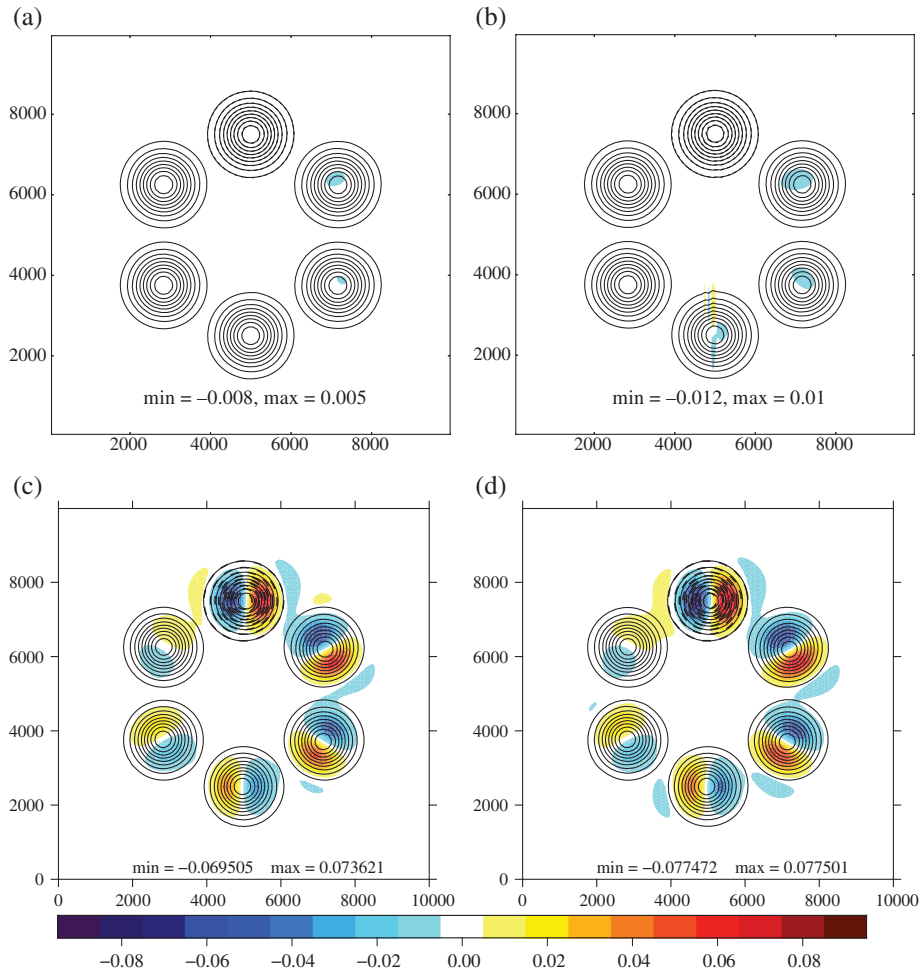


Figure 3. Results of the solid-body rotation test case on (a, c) orthogonal and (b, d) non-orthogonal meshes every 100 s for meshes of 100×100 cells and $\Delta t = 1$ s, giving a maximum Courant number of around 1. (a, b) show the dimensionally split scheme, and (c, d) the multi-dimensional scheme. The contours show the tracer value every 0.1 from 0.1 to 0.9 and the colours show the errors. The dashed contour shows the initial conditions. The minimum and maximum errors are given in each panel.

mountain range defined by

$$h = \begin{cases} h_0 \cos^2 \frac{\pi x}{\lambda} \cos^2 \frac{\pi x}{2a} & \text{for } |x| \leq a, \\ 0 & \text{otherwise,} \end{cases} \quad (37)$$

with the maximum mountain height, $h_0 = 3$ km, half-width $a = 25$ km and wavelength $\lambda = 8$ km. These values give a maximum terrain gradient of close to 45° . The wind is given by a streamfunction which is defined at vertices so that the wind field is discretely divergence-free. The streamfunction at vertices is calculated analytically from the wind profile:

$$u(z) = u_0 \begin{cases} 1 & \text{for } z_2 \leq z, \\ \sin^2 \left(\frac{\pi}{2} \frac{z - z_1}{z_2 - z_1} \right) & \text{for } z_1 < z \leq z_2, \\ 0 & \text{for } z < z_1, \end{cases} \quad (38)$$

with $u_0 = 10 \text{ m s}^{-1}$, $z_1 = 4$ km and $z_2 = 5$ km. The initial tracer position is given by:

$$\phi = \begin{cases} \cos^2 \frac{\pi r}{2} & \text{for } r \leq 1, \\ 0 & \text{otherwise,} \end{cases} \quad (39)$$

$$\text{with } r = \sqrt{\left(\frac{x - x_0}{A_x} \right)^2 + \left(\frac{z - z_0}{A_z} \right)^2},$$

initial tracer centre $(x_0, z_0) = (-50 \text{ km}, 9 \text{ km})$ and halfwidths $A_x = 25 \text{ km}$, $A_z = 9 \text{ km}$. At time $t = 5000$ s the tracer is above

the mountain and the simulation finishes at $t = 10\,000$ s by which time the analytic solution is centred at $(50, 8 \text{ km})$.

The tracer advection over orography is shown in Figure 7 for the split and multi-dimensional schemes at a resolution of $\Delta x = 1 \text{ km}$, $\Delta z = 500 \text{ m}$ and for a range of Courant numbers. The multi-dimensional scheme uses implicit time-stepping. The horizontal Courant number is defined as $u_0 \Delta t / \Delta x$ and ranges from 0.25 to 10. The maximum Courant number is the maximum of the multi-dimensional Courant number which is defined for cell with faces f as

$$c = \frac{1}{2V} \sum_f |\mathbf{u}_f \cdot \mathbf{S}_f| \Delta t \quad (40)$$

(section 2.3 defines the variables), and ranges from 0.74 to 29.6. The time-step restriction for the split scheme is based on the deformational Courant number, $c_d \leq 1$, (Eq. (11)). The maximum deformational Courant number is also given in Figure 7. The contours in Figure 7 show the tracer values at 0, 5 000 s and 10 000 s after initialization and the colours show the errors from the analytic solution. For horizontal Courant numbers less than 1 (maximum Courant number up to 3), both schemes give accurate results with the dimensionally split scheme tending to give oscillations and the multi-dimensional scheme producing more diffusion. For larger Courant numbers, when the deformational Courant number is greater than 1, the split scheme is unstable, while the multi-dimensional scheme produces large phase errors due to the errors associated with implicit time-stepping. The term responsible for the large deformational Courant number is $\partial u / \partial z$ where the velocity shears from $u_0 = 10 \text{ m s}^{-1}$ at $z = z_2$ to zero at $z = z_1$.

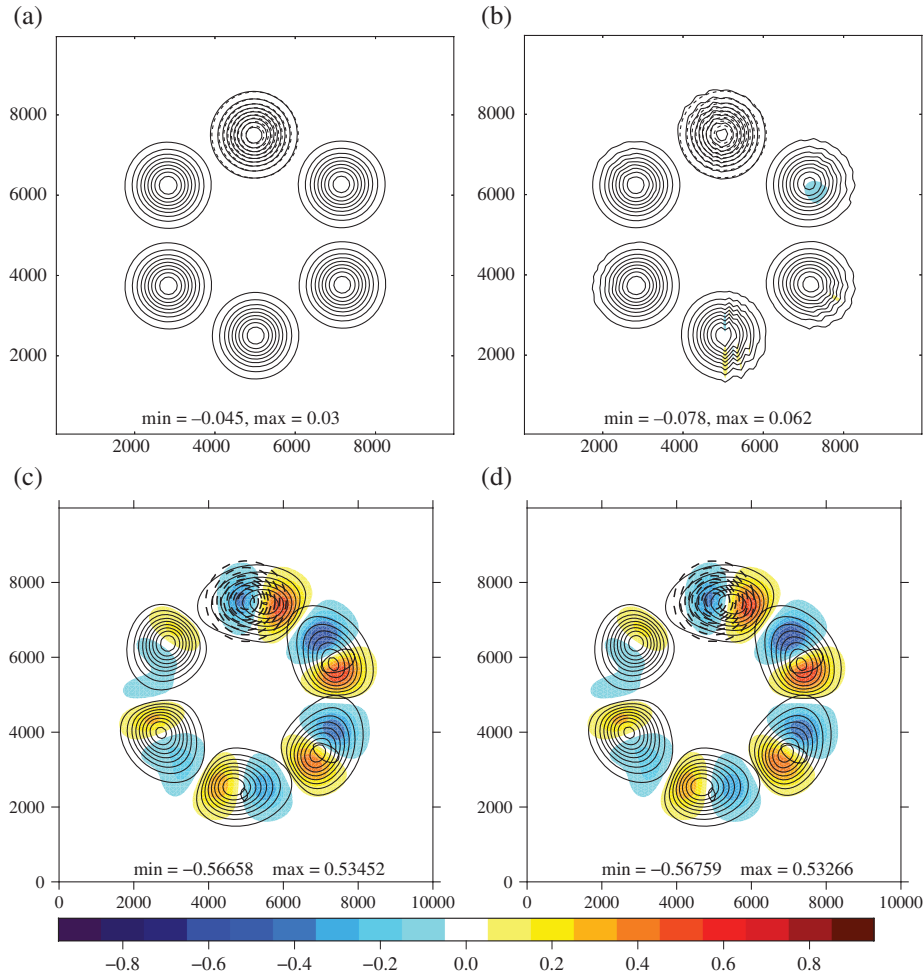


Figure 4. As Figure 3, but for $\Delta t = 10$ s, giving a maximum Courant number of around 10.

We examine the convergence with resolution in Figure 8 which shows the ℓ_2 and ℓ_∞ error norms as a function of Δx . These simulations all use a maximum Courant number less than 1, a horizontal Courant number of 0.25, a maximum deformational Courant number of about 0.2 and fixed ratios of Δx , Δz and Δt . The width and height of the orography is kept constant. Both schemes give similar accuracy with the dimensionally split scheme having faster convergence with resolution.

In summary, the dimensionally split scheme has good accuracy over orography for modest Courant numbers but larger errors for larger Courant numbers, and is unstable when the deformational Courant number is greater than 1, whereas the multi-dimensional scheme with implicit time-stepping is stable for all Courant numbers. The multi-dimensional scheme is second-order convergent and the dimensionally split scheme converges faster.

3.3. Deformational flow

For deformational flow, there is no analytical solution and therefore we follow the approach of Nair and Lauritzen (2010) and Lauritzen *et al.* (2012) and define an evolving velocity field that reverses direction half-way through the simulation, taking the tracer back to the initial conditions. Error norms can then be calculated by comparing the final and initial tracer fields. The deformational velocity field is added to a fixed solid-body rotation that does not reverse so as to avoid error cancellation between the forward and backward periods (solid body here meaning horizontal wind on a periodic domain).

We define a Cartesian version of the deformational, non-divergent test case of Lauritzen *et al.* (2012) with a domain between $-\pi$ and π in the x direction ($L_x = 2\pi$) and between $-\pi/2$ and $\pi/2$ in the y direction ($L_y = \pi$) with periodic boundary

conditions in the x direction and zero gradient, and zero flow boundary conditions in the y direction. The streamfunction adapted to Cartesian co-ordinate is:

$$\psi = \frac{\hat{\psi}}{T} \left(\frac{L_x}{2\pi} \right)^2 \sin^2 \left\{ 2\pi \left(\frac{x}{L_x} - \frac{t}{T} \right) \right\} \cos^2 \left(\frac{\pi y}{L_y} \right) \cos \left(\frac{\pi t}{T} \right) - \frac{L_x y}{T}, \quad (41)$$

where $\hat{\psi} = 10$ and $T = 5$ is the time for one complete revolution of the periodic domain for the solid-body rotation part of the flow. In order to test the order of convergence of the advection schemes, we use the infinitely smooth Gaussian distribution for the initial tracer concentration:

$$\phi = 0.95 \exp \left(-\frac{|\mathbf{x} - \mathbf{x}_0|^2}{A} \right) + 0.95 \exp \left(-\frac{|\mathbf{x} - \mathbf{x}_1|^2}{A} \right), \quad (42)$$

where $\mathbf{x}^T = (x, y)$, $\mathbf{x}_0^T = (5L_x/12, 0)$, $\mathbf{x}_1^T = (7L_x/12, 0)$ and $A = 1/5$.

A mesh with distortion similar to a cubed-sphere is defined by the co-ordinate transform:

$$X = x, \quad Y = \begin{cases} L_y \frac{y-f}{L_y-2f} & \text{for } y \geq f, \\ L_y \frac{y+f}{L_y+2f} & \text{for } y < f, \end{cases} \quad (43)$$

where f is:

$$f = \begin{cases} \frac{1}{\sqrt{3}} \left(\frac{\pi}{4} - |x| \right) & \text{for } |x| \leq \frac{\pi}{2}, \\ \frac{1}{\sqrt{3}} \left(|x| - \frac{3\pi}{4} \right) & \text{for } |x| > \frac{\pi}{2}. \end{cases}$$

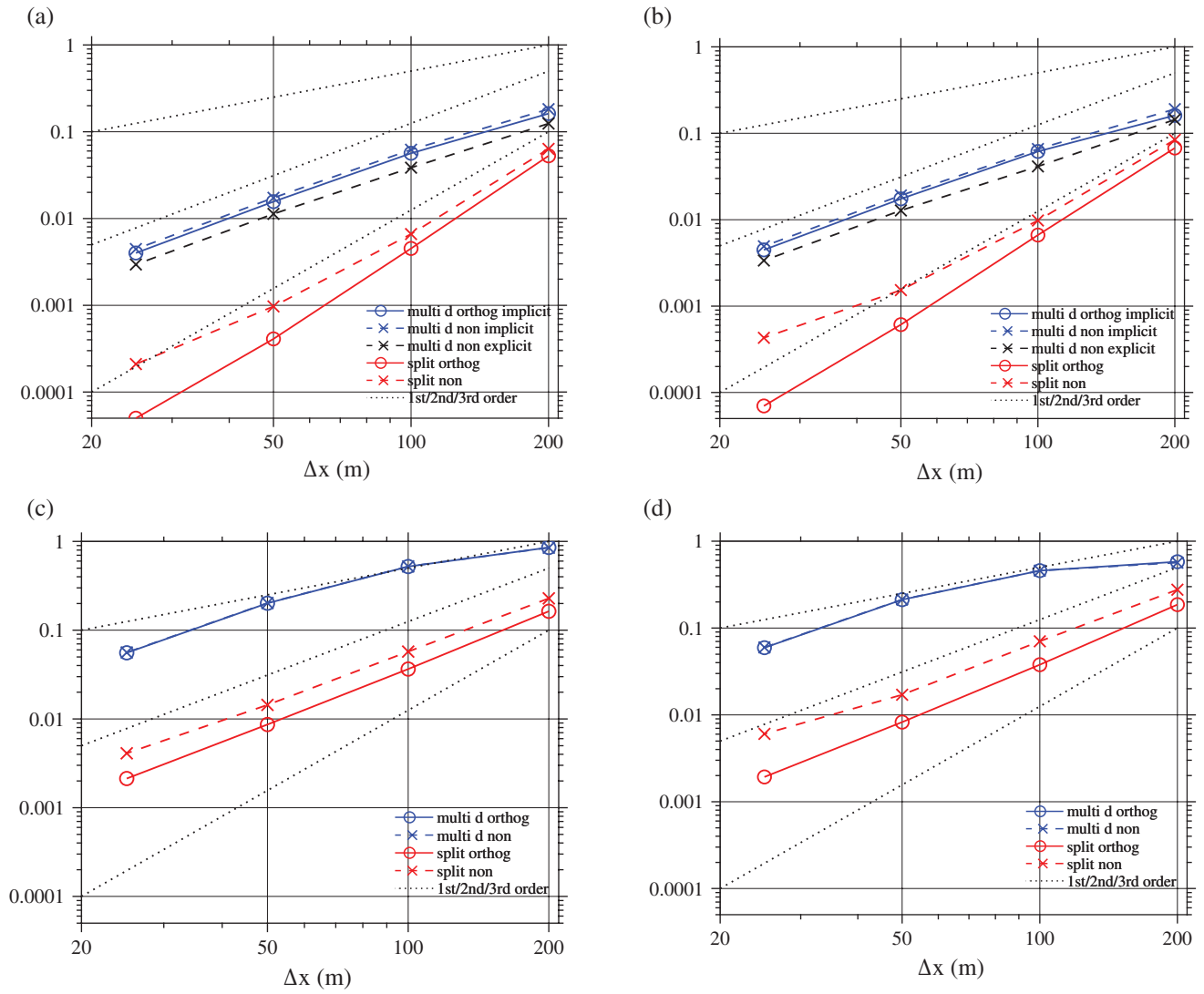


Figure 5. Convergence with spatial resolution of the (a,c) ℓ_2 and (b,d) ℓ_∞ errors for the solid-body rotation test case at $t = 500$ s (5/6 of one revolution) on orthogonal and non-orthogonal meshes using multi-dimensional and dimensionally split schemes. (a, b) are for $c \approx 1$ and (c, d) for $c \approx 10$.

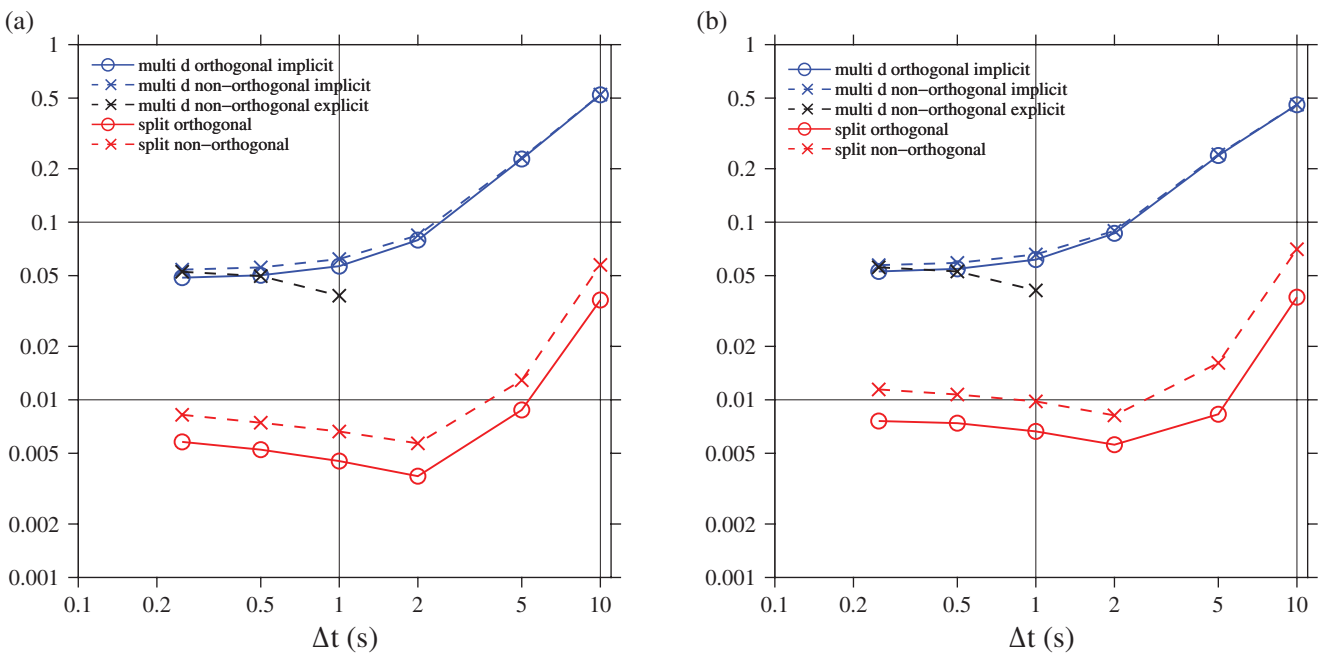


Figure 6. Convergence with temporal resolution of the (a) ℓ_2 and (b) ℓ_∞ errors for the solid-body rotation test case at $t = 500$ s (5/6 of one revolution) using orthogonal and non-orthogonal meshes of 100×100 cells using multi-dimensional and dimensionally split schemes. Maximum Courant numbers range from $c = 0.25$ (for $\Delta t = 0.25$ s) to $c = 10$ (for $\Delta t = 10$ s). The multi-dimensional scheme shows results for implicit and explicit time-stepping on the non-orthogonal mesh.

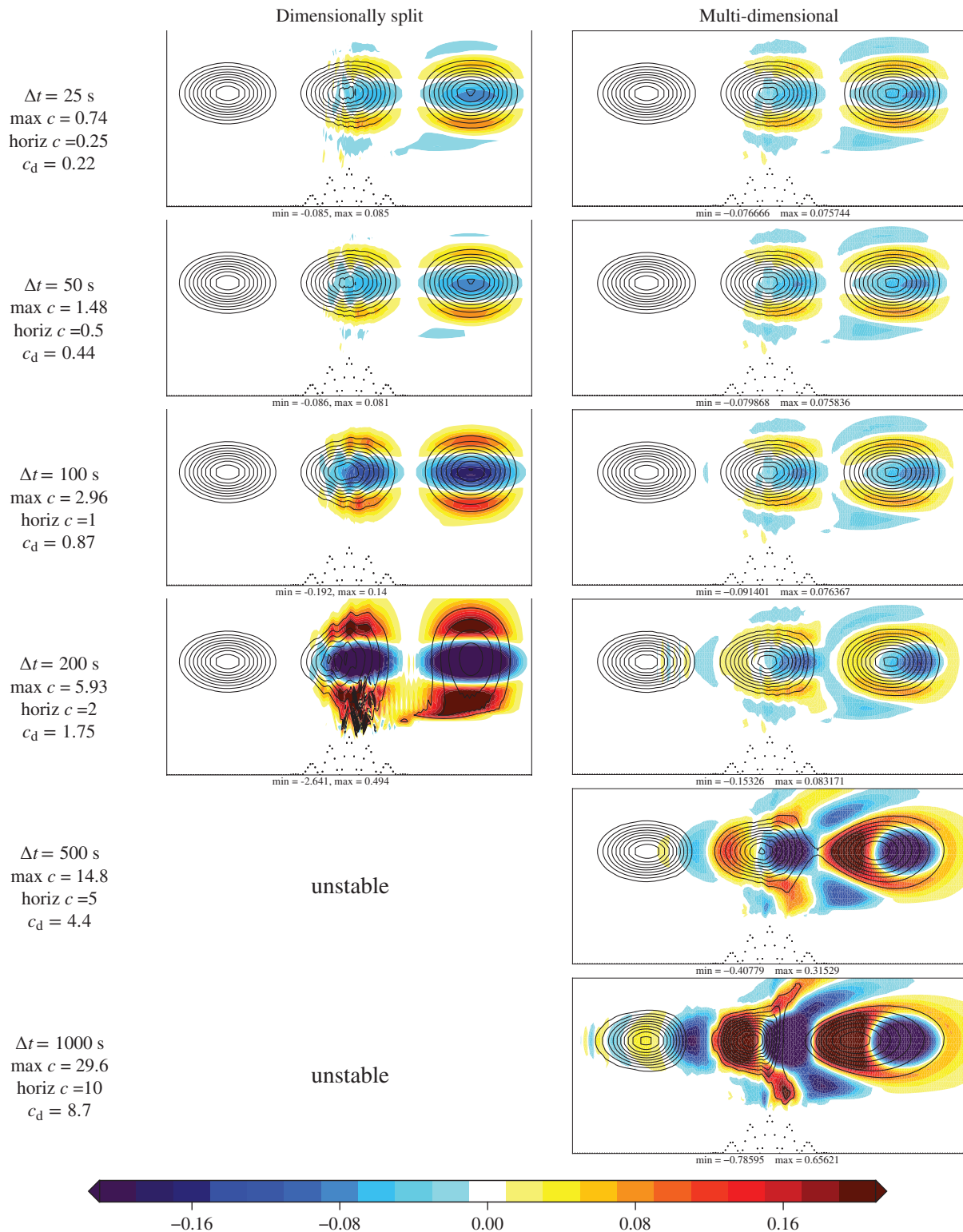


Figure 7. Tracer contours after 0, 5000 and 10000 s for the horizontal advection over orography, with errors in colour; each row shows a different time step. Spatial resolution is $\Delta x = 1$ km, $\Delta z = 500$ m. Maximum Courant number, horizontal Courant number, c , and maximum deformational Courant number, c_d are also given.

The initial conditions and a 120×60 mesh with distortion defined by this co-ordinate transform is shown in Figure 9.

The tracer concentrations after 1, 2, 3, 4 and 5 time units are shown in Figure 10 using the dimensionally split and multi-dimensional schemes (with explicit time-stepping, although the results using implicit time-stepping look identical) on a non-orthogonal mesh of 480×240 cells, a time step of 0.0025 units (ie 2000 time steps to reach 5 time units) which gives a maximum Courant number of 1.03. The tracer is stretched out, wound up, advected around and then wound back into its original position with some numerical errors. Both advection schemes preserve fine filaments but suffer from some dispersion errors which generate small oscillations around zero behind sharp gradients in

the direction of the flow since neither scheme is monotonic or positive preserving. The dimensionally split scheme returns the tracer to a more accurate final solution.

Sensitivity to orthogonality, resolution, time step and time scheme are explored using a range of simulations with maximum Courant numbers shown in Table 1. The Courant numbers on the uniform, orthogonal meshes (with no mesh distortions) are 70% of those on the non-orthogonal mesh since the non-orthogonal mesh has clustering of mesh points. The deformational Courant number is less than 1 for all simulations. The convergence with resolution of the ℓ_2 and ℓ_∞ error norms at the final time are shown in Figure 11 for both advection schemes using a maximum Courant number close to 1 and for a maximum Courant number

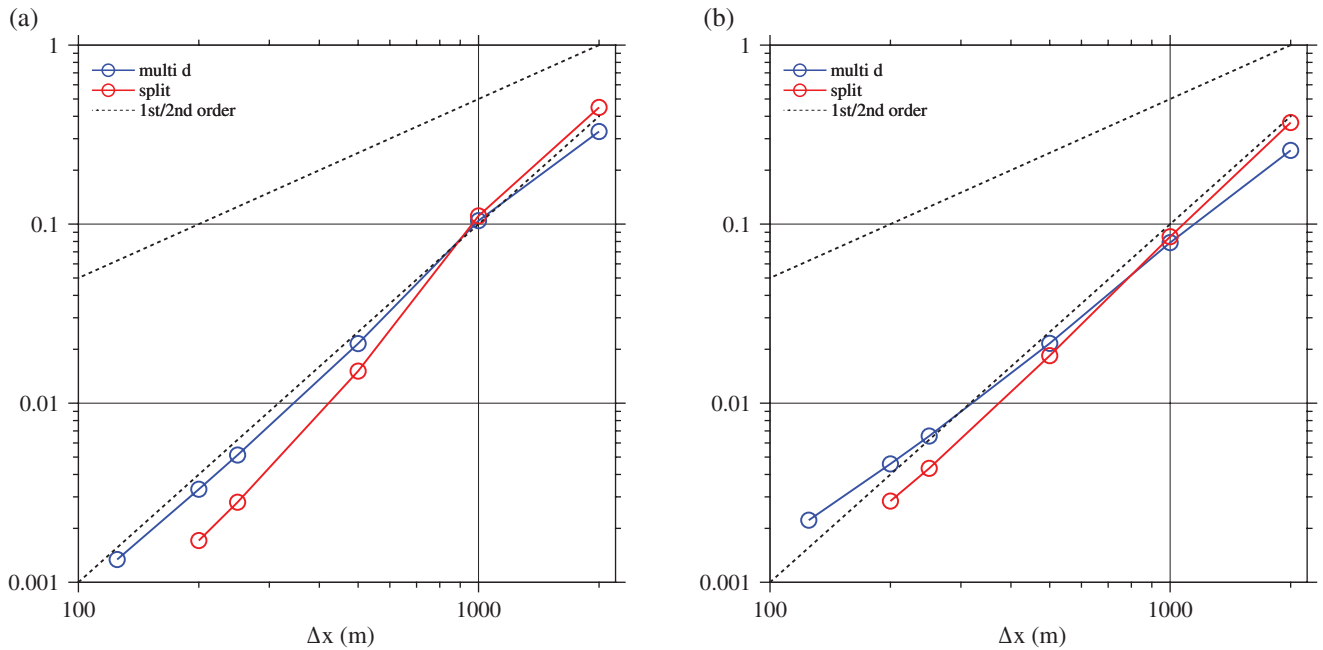


Figure 8. Convergence of the (a) ℓ_2 and (b) ℓ_∞ error metrics for horizontal advection over a mountain after 10 000s for the dimensionally split (PPM) scheme and the implicit multi-dimensional MOL scheme, all using a horizontal Courant number of 0.25. Dotted lines show theoretical first- and second-order convergence.

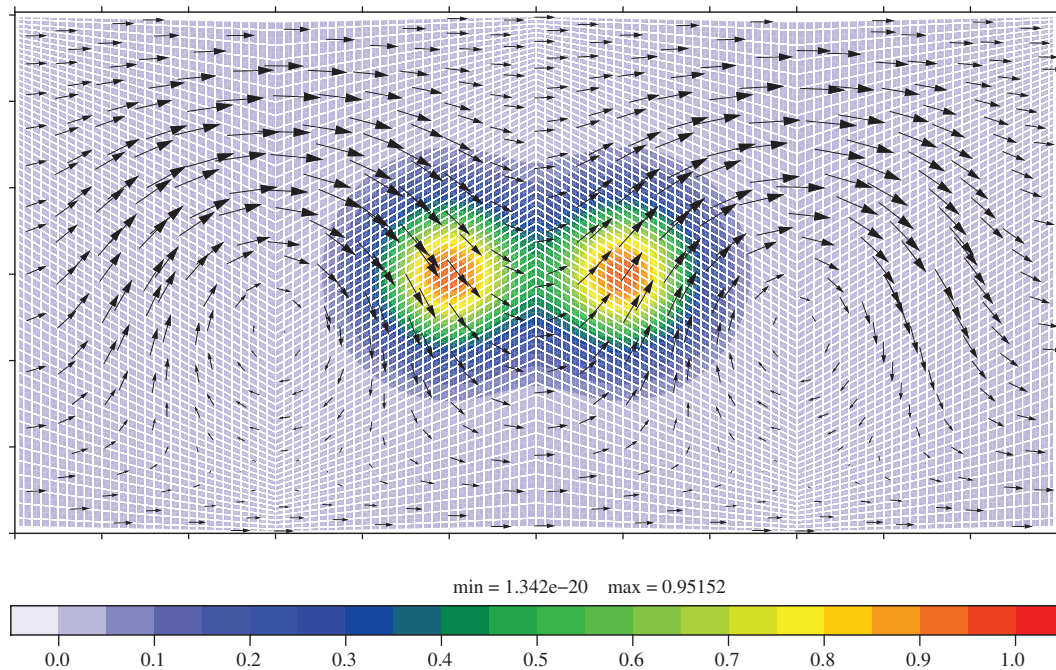


Figure 9. The non-orthogonal mesh/co-ordinate system for the deformational flow with 120×60 cells and the initial tracer conditions.

close to 10. We will first consider the behaviour of the schemes at modest Courant number (maximum close to 1). The explicit multi-dimensional scheme is stable at this Courant number and errors are almost identical to those of the implicit scheme. All of the schemes give first-order convergence with resolution at coarse resolutions due to error saturation (if the scheme is stable, errors are bounded above by about 1). At higher resolution, the split scheme converges with nearly third-order for both meshes whereas the multi-dimensional scheme approaches second-order. For large Courant numbers (maximum close to 10), the split scheme converges with first-order for both mesh types due to the crude estimation of the departure point (section 2.1), whereas the multi-dimensional scheme is much less accurate but approaches second-order at high resolution. The dimensionally split scheme is sensitive to mesh distortions only for large Courant numbers.

We inspect sensitivity to time step of all schemes on orthogonal and non-orthogonal meshes of 120×60 cells in Figure 12 using

the time steps shown in Table 1 giving maximum Courant numbers ranging from 0.25 to 10 (deformational Courant numbers always less than 0.32). Again this shows that the explicit time-stepping gives almost identical errors to the implicit time-stepping for time steps at which it is stable. Figure 12 demonstrates potentially useful properties of the FFSL time-stepping used by the split scheme. Ignoring errors in calculating the departure point, semi-Lagrangian schemes have errors proportional to Δt^{-1} (Durrant, 2010) which explains the reduction in error as time step increases for the split scheme. Once the maximum Courant number reaches 1 or 2 ($\Delta t = 0.01$ or 0.02), errors of the split scheme do grow with the time step, due to the deformational nature of the flow and first-order errors in calculating the departure points. In contrast, using the multi-dimensional scheme which uses method-of-lines time-stepping, errors increase as the time step is increased. Comparisons between schemes in Figures 10 and 11 used maximum Courant numbers of 1 and 10 which showed that the split scheme on the orthogonal

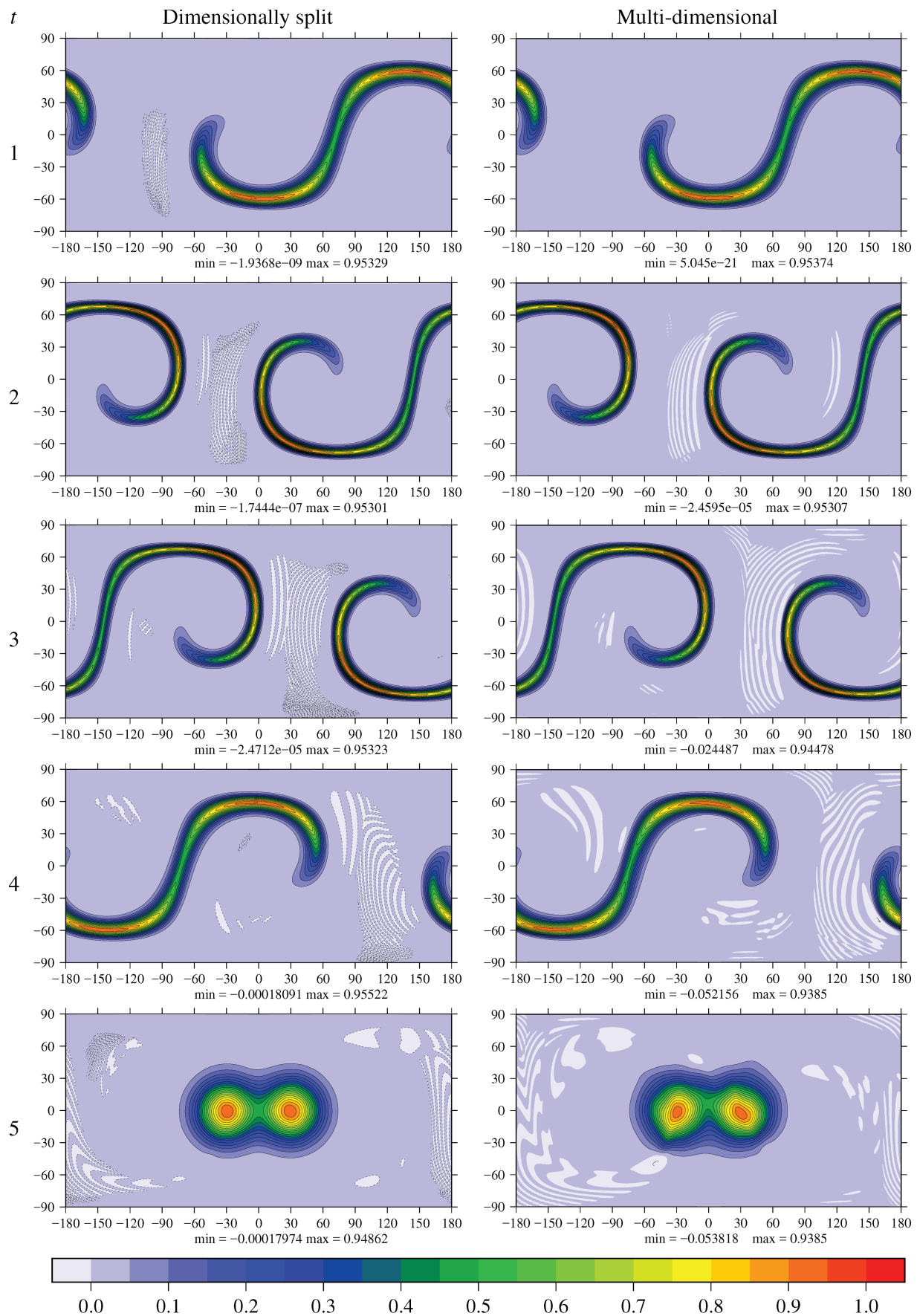


Figure 10. Results of the deformational flow test case after 1, 2, 3, 4 and 5 time units on the non-orthogonal meshes of 480×240 cells using the explicit dimensionally split (left) and multi-dimensional (right) schemes.

mesh gave better accuracy than the multi-dimensional scheme. However, Figure 12 shows that this advantage disappears at lower Courant numbers since the multi-dimensional scheme (method-of-lines) gets more accurate with lower Courant numbers, whereas the split scheme (semi-Lagrangian) gets less accurate since more

time steps have to be taken. Both schemes are stable for all time steps considered.

In summary, the dimensionally split scheme has good accuracy for deformational flow independent of mesh orthogonality. The multi-dimensional scheme is competitive at small Courant

Table 1. Resolutions, time steps and Courant numbers for the non-orthogonal meshes for the deformational flow test case.

Resolution	60 × 30	120 × 60	240 × 120	480 × 240	960 × 480	1920 × 960
Δt	Maximum Courant numbers (deformational)					
0.2	10.3 (0.639)	–	–	–	–	–
0.1	–	10.3 (0.312)	–	–	–	–
0.05	–	5.17 (0.154)	10.3 (0.154)	–	–	–
0.025	–	–	–	10.3 (0.077)	–	–
0.02	–	2.06 (0.061)	–	–	–	–
0.0125	1.03	–	–	–	10.3	–
0.01	–	1.03 (0.030)	–	–	–	–
0.00625	–	–	–	–	–	10.3
0.005	–	0.517 (0.015)	1.03 (0.015)	–	–	–
0.0025	–	0.254 (0.008)	–	1.03 (0.008)	–	–
0.00125	–	–	–	–	1.03 (0.004)	–
0.000625	–	–	–	–	–	1.03

Maximum Courant numbers are shown inside the table for simulations which are presented in graphs. The maximum Courant numbers on the orthogonal meshes are 70% of those on the non-orthogonal meshes. The maximum deformational Courant numbers are in brackets for some cases

numbers but the semi-Lagrangian nature of the split scheme means that errors are very low when the Courant number is close to 1.

3.4. Computational cost

We cannot compare CPU time, wall-clock time or parallel efficiency of the two advection schemes because the multi-dimensional scheme is written in C++ using OpenFOAM and the split scheme is written in Python; the two codes have been run on different hardware and the split advection scheme code has not been parallelised. Instead we consider the number of multiply operations performed per cell per time step for calculating the fluxes of each tracer. We appreciate that this is not a good predictor of efficiency as it does not consider memory read and write requirements or cache coherency. However all data that are multiplied have to be fetched from memory and so, assuming that all data can be arranged optimally in memory to enable fewest cache misses, the number of multiplies should be related to the wall-clock time. Neither advection scheme does significantly more work per memory fetch than the other.

For both schemes we consider the number of multiplies needed to calculate the flux of ϕ at each face, i.e. Eq. (8) for the dimensionally scheme and Eq. (24) for the multi-dimensional scheme. We do not consider the computational cost of updating the cell averages from the fluxes using Gauss's theorem (Eqs (3) and (22)) as these are the same for each scheme. For the multi-dimensional scheme, we do include an estimate of the amount of work done by the linear equation solver, but we do not consider the scalability of this solver.

3.4.1. Dimensionally split scheme

The computational cost of PPM with COSMIC splitting is not strongly time-step-dependent due to the swept area approach of FFSL. One-dimensional PPM uses four cells for interpolation to find face values used by the reconstruction. Assuming as much as possible is pre-computed, this interpolation uses three multiplies on a non-uniform mesh. The reconstruction then uses six multiplies to calculate the flux on each face. Only one additional memory access and one additional multiply are needed per cell for Courant numbers greater than 1 (Eq. 9) making ten multiplies per cell for applying PPM in one direction for a Courant number greater than 1. The COSMIC splitting in two dimensions involves four applications of the one-dimensional PPM. This makes 40 multiplies per cell in total for applying PPM with COSMIC splitting in two dimensions. In three dimensions, COSMIC splitting requires 12 applications of PPM (Leonard *et al.*, 1996) leading to 120 multiplies.

3.4.2. Multi-dimensional scheme

The number of multiplies involved in the multi-dimensional scheme includes the number of multiplies to update the higher-order advection and the number of multiplications for each iteration of the linear equation solver. We will also consider the cost of an explicit version of the multi-dimensional scheme using an RK2 time-stepping scheme (e.g. as used by Shaw *et al.*, 2017) which is stable and accurate up to a Courant number of 1 or 2 for this spatial discretization and gives very similar results to the implicit scheme.

On a logically rectangular two-dimensional mesh, there are 12 cells in each stencil for each face (Figure 1). Each cell has four faces and the interpolation onto each face is used to calculate the flux between two cells. This leads to 24 multiplies per cell for each RK2 stage and hence 48 multiplies per cell per time step. When using implicit time-stepping, there will be 24 multiplies per cell for every evaluation of the right-hand side of the matrix for the higher-order correction on first-order upwind.

We have not explored the sensitivity of the accuracy and stability to the stencil size and shape in three dimensions. The three-dimensional equivalent of the stencil of quadrilaterals in Figure 1 is likely to contain 36 (rather than 12) cells, although it may be possible to omit some corner cells and use a stencil of 20 cells. For a stencil of 36 cells, the number of multiplies per cell per time step would be $36 \times 3 \times 2 = 216$.

The implicit version of the multi-dimensional scheme (section 2.3.1) requires the solution of an asymmetric, diagonally dominant matrix with three non-zero elements per row for a logically rectangular two-dimensional mesh. The pre-conditioner is implemented in file DILUPreconditioner.C in OpenFOAM 3.0.1 and the solver in file PBiCG.C. From these files, we estimate that, for a mesh of quadrilaterals, the solver will use 24 multiplies (or divides) per cell, per solver iteration, including pre-conditioning. The average number of iterations of the preconditioned bi-CG solver per time step for each of the simulations is shown in Table 2 (including the number of solver iterations in each outer iteration). These simulations use two outer iterations per time step when the maximum Courant number is ≤ 1.1 (as in Eqs (27) and (28)) but, for stability, use four outer iterations per time step for the larger Courant numbers. This partly explains the greater number of iterations for larger Courant numbers. A solver tolerance of 10^{-8} is used for each of the outer solves. Table 2 shows that the total number of iterations per time step reduces slightly as resolution increases. The total number of iterations for a complete simulation is reduced by using larger Courant numbers because the number of iterations per time step increases less than linearly with increasing Courant number. In fact, simulations with larger Courant numbers are considerably cheaper because there are fewer evaluations of the right-hand side of the matrix equation.

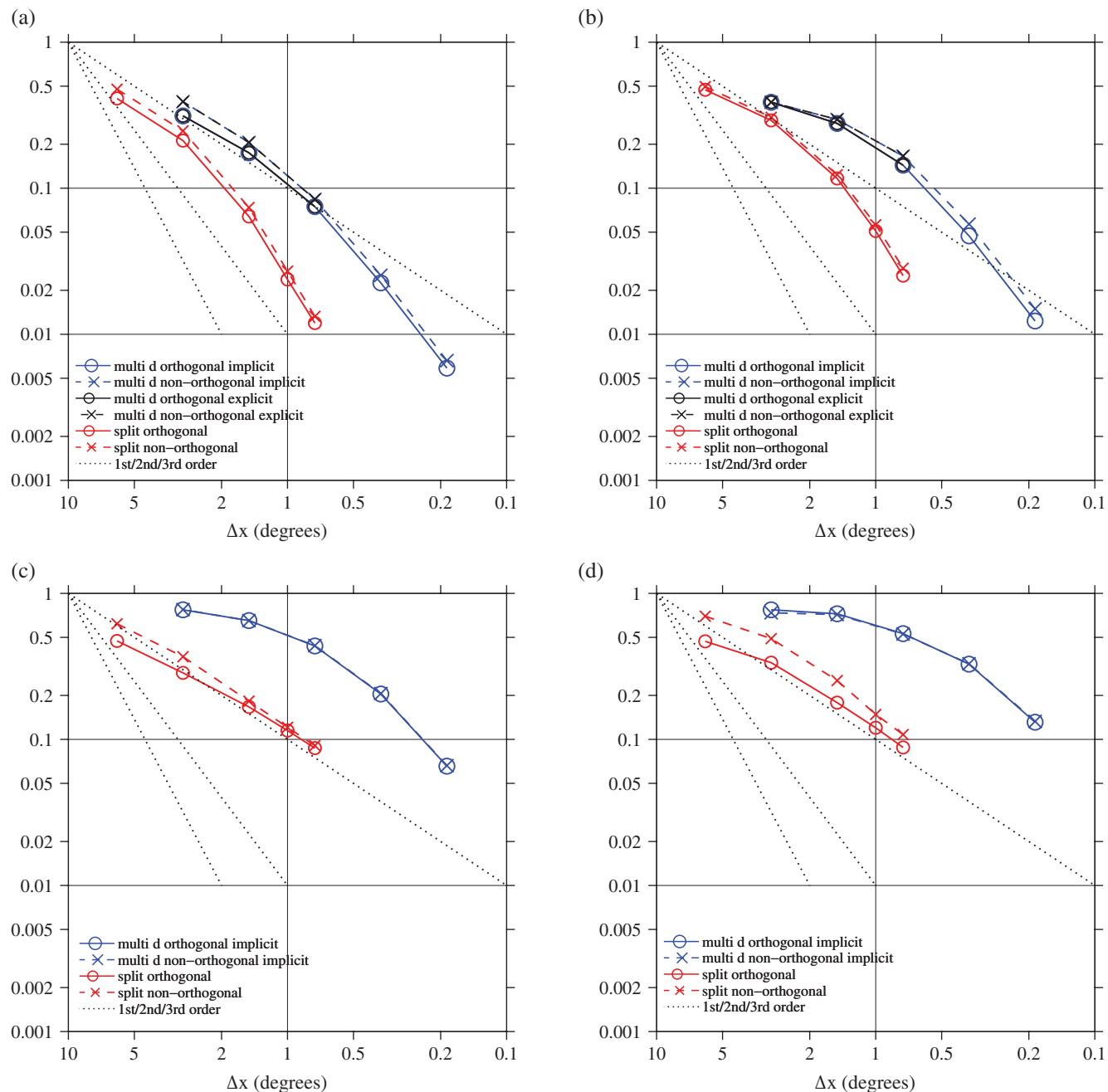


Figure 11. Convergence with resolution of the (a,c) ℓ_2 and (b,d) ℓ_∞ errors for the deformational flow on orthogonal and non-orthogonal meshes using multi-dimensional and dimensionally split schemes. (a,b) are for $c \leq 1$ and (c,d) for $c \leq 10$.

Table 2. Number of solver iterations per time step of the multi-dimensional scheme for all the simulations on orthogonal/non-orthogonal meshes for the deformational flow test case.

Resolution	120 × 60	240 × 120	480 × 240	960 × 480	1920 × 960
Max c	Iterations per time step (orthogonal/non-orthogonal)				
10	33.4/39.5	31.3/38.0	26.0/32.8	19.3/25.0	13.5/18.6
5	22.4/25.7	—	—	—	—
2	13.1/14.6	—	—	—	—
1	6.0/ 6.5	5.4/ 5.8	4.9/ 5.3	4.6/ 4.9	3.9/ 4.2
0.5	4.9/ 5.0	—	—	—	—
0.25	3.9/ 3.9	—	—	—	—

3.4.3. Cost comparisons

Combining the number of solver iterations, the number of multiply operations per solve and the number of multiply operations in calculating the explicit higher-order part of the advection scheme, the total number of multiply operations per cell per time step for the multi-dimensional scheme is shown

in Table 3 for the non-orthogonal mesh of 120×60 cells for the deformational flow test case. Table 3 also shows the number of multiplies for using the explicit, RK2 version of the multi-dimensional scheme and the dimensionally split scheme.

Table 3 shows that the implicit scheme always uses more multiply operations but particularly uses more multiplies for large Courant numbers. The explicit version of the multi-dimensional scheme always uses 48 multiply operations but is not stable for all time steps whereas the dimensionally split scheme (using FFSL time-stepping) is stable for all Courant numbers (at this spatial resolution) and always uses the fewest number of multiply operations per cell per time step.

There is considerable flexibility in the solver configuration: the number of outer iterations per time step determines how frequently the high-order correction on the right-hand side of the matrix equation is updated, and the solver tolerance per outer iteration could be modified by using a weaker tolerance on all but the final matrix solve per time step. These options have not been explored. It may also be beneficial to create more non-zero matrix entries rather than having the higher-order correction entirely a deferred correction on first-order upwind, but such a

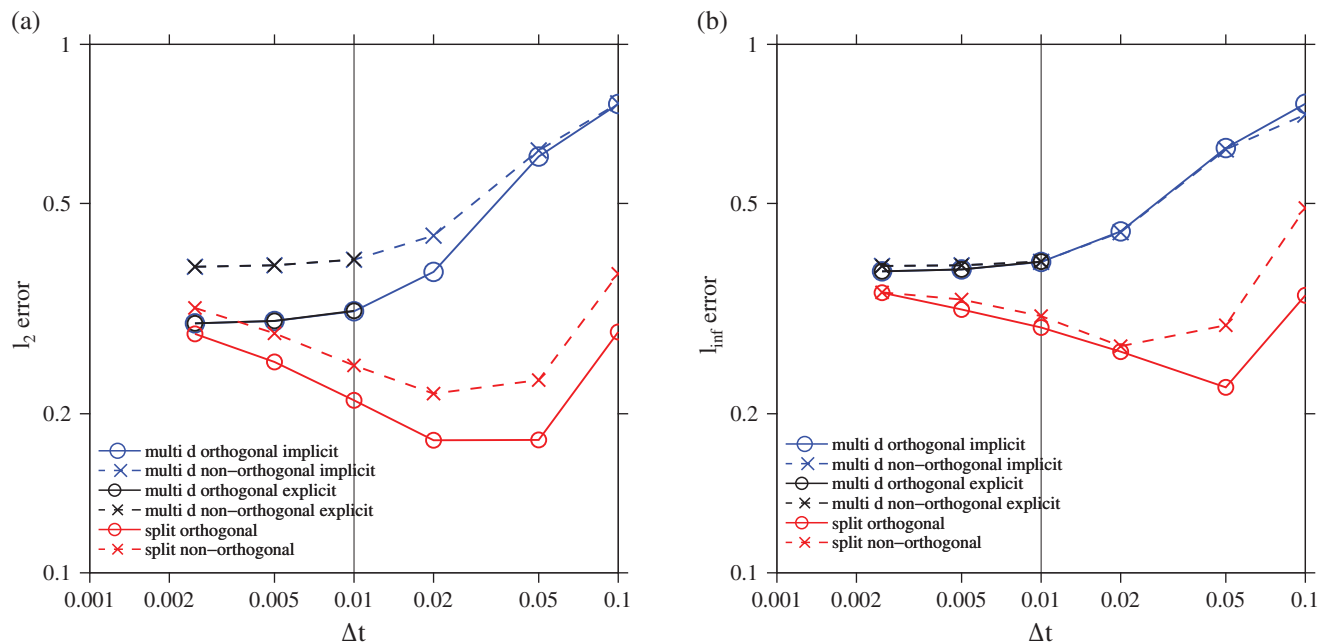


Figure 12. Variation of (a) ℓ_2 and (b) ℓ_∞ errors with time step for the deformational flow on orthogonal and non-orthogonal meshes of 120×60 cells using multi-dimensional and dimensionally split schemes. Different time steps give maximum Courant numbers of 0.25, 0.5, 1, 2, 5 and 10.

Table 3. Total number of multiply operations per cell per time step for different Courant numbers for the multi-dimensional scheme, an explicit version of the multi-dimensional scheme (using RK2 or Heun time-stepping) and the dimensionally split scheme on two-dimensional, logically rectangular meshes for the deformational flow test case.

Max c	Multi-dimensional		Dim. split
	Implicit	Explicit	
10	$39.5 \times 24 + 48 \times 2 = 1\,044$	—	40
5	$25.7 \times 24 + 48 \times 2 = 712.8$	—	40
2	$14.6 \times 24 + 48 \times 2 = 446.4$	—	40
1	$6.5 \times 24 + 48 = 204$	48	40
0.5	$5.0 \times 24 + 48 = 168$	48	40
0.25	$3.9 \times 24 + 48 = 141.6$	48	40

change would need to ensure that the matrix remains diagonally dominant.

We also measured the wall-clock computation cost, comparing the implicit and explicit versions of the multi-dimensional scheme running the deformational test case in serial on a non-orthogonal mesh with a maximum Courant number of 1. At the resolution of 120×60 , the model using implicit time-stepping is 5% more expensive than the explicit model. This drops to 2.5% at the resolution of 960×480 . Despite the implicit scheme having four times as many multiplies as calculated for Table 3, the difference in wall-clock time is negligible. This demonstrates both the danger of only looking at the number of multiplies and the danger of using results of timing code. The linear equation solvers on OpenFOAM have undergone years of optimization whereas the stencils for calculating the cubic fit have not been optimized and entail unpredictable memory access patterns and likely frequent cache misses.

4. Summary and conclusions

We examine the errors associated with using a dimensionally split advection scheme and a multi-dimensional advection scheme on distorted meshes. The dimensionally split scheme is the piecewise polynomial method (PPM; Colella and Woodward, 1984) with COSMIC splitting (Leonard *et al.*, 1996) which extends it to two spatial dimensions. PPM converges with third-order in one dimension and COSMIC splitting enables second-order convergence in two dimensions. PPM is a FFSL

scheme and so can handle large Courant numbers accurately without significant additional computational cost, with a time-step restriction based on the deformational Courant number. The second-order accurate multi-dimensional scheme is split in space and time (method-of-lines) and uses a cubic polynomial fit over a stencil of cells for spatial discretisation and explicit RK2 or Crank–Nicolson in time to retain stability for large Courant numbers. The multi-dimensional scheme uses cell centroids as Gauss points so the order of accuracy is limited to two. We use versions of both schemes without any monotonicity constraints in order to compare the handling of multi-dimensionality of the two schemes and order of convergence rather than comparing the limiters of the two schemes. When run explicitly, the cost of the two schemes is similar. These schemes are therefore considered suitable for comparison.

Three two-dimensional advection test cases on Cartesian planes are proposed without the complexities of a spherical domain or multi-panel meshes but with distorted meshes to mimic the distortions near cubed-sphere edges or the distortions of terrain-following co-ordinates. We therefore propose that these test cases could be used in the initial testing of advection schemes, before the generation of meshes on the sphere. The first test case is an extension of the Leonard *et al.* (1996) solid-body rotation using a distorted mesh. The second test case is the established horizontal advection over orography (Schär *et al.*, 2002) using a basic terrain-following mesh in order to maximize the distortions due to terrain-following coordinates. The third test case is the deformational flow test case of Lauritzen *et al.* (2012) adapted to a planar, Cartesian domain and a distorted mesh. We use the version of this test case with smooth initial conditions (the sum of two Gaussians) in order to examine order of convergence.

The dimensionally split scheme is most accurate on orthogonal meshes but only loses a little accuracy on highly distorted meshes, despite a first-order departure point calculation. The multi-dimensional scheme is almost entirely insensitive to mesh distortion and asymptotes to second-order convergence at high resolution. As is expected for implicit time-stepping, phase errors occur when using long time steps but the spatially well-resolved features are advected at the correct speed and the multi-dimensional scheme is always stable. On orthogonal meshes, the dimensionally split scheme remains highly accurate when large Courant numbers are used (as long as the deformational Courant number is low). However, on distorted meshes, the split scheme loses accuracy at large Courant numbers.

Computational cost is estimated by the number of multiplies. The matrix solver associated with the implicit time-stepping of the multi-dimensional scheme means that it is always more expensive than the split scheme with cost increasing with Courant number. We have not investigated sensitivity to solver and pre-conditioner choices and it may be possible to do better. However in practice we found that the implicit solution contributed only 2.5–5% of the total wall-clock computation time. The FFSL method enables long time steps without any matrix solutions and this will always be difficult to beat. We also use the multi-dimensional scheme with explicit time-stepping, making the cost similar to the dimensionally split scheme.

The dimensionally split scheme is hard to beat, providing close to third-order convergence, even in the presence of mesh distortions, and can be very cheaply extended for large Courant numbers.

References

- Bott A. 2010. Improving the time-splitting errors of one-dimensional advection schemes in multidimensional applications. *Atmos. Res.* **97**: 619–631.
- Brassington G, Sanderson B. 1999. Semi-Lagrangian and COSMIC advection in flows with rotation or deformation. *Atmos. Ocean* **37**: 369–388.
- Colella P, Woodward P. 1984. The piecewise parabolic method (PPM) for gas-dynamical simulations. *J. Comput. Phys.* **54**: 174–201.
- Davies T, Cullen MJP, Malcolm A, Mawson M, Staniforth A, White AA, Wood N. 2005. A new dynamical core for the Met Office's global and regional modelling of the atmosphere. *Q. J. R. Meteorol. Soc.* **131**: 1759–1782. <https://doi.org/10.1256/qj.04.101>.
- Dennis JM, Edwards J, Evans KJ, Guba O, Lauritzen PH, Mirin AA, St-Cyr A, Taylor MA, Worley PH. 2012. CAM-SE: A scalable spectral element dynamical core for the community atmosphere model. *Int. J. High Perform. Comput. Appl.* **26**: 74–89.
- Durran D. 2010. *Numerical Methods for Fluid Dynamics with Applications to Geophysics*, (2nd edn), Texts in Applied Mathematics 32. Springer: Berlin.
- Durran D, Blossey P. 2012. Implicit-explicit multistep methods for fast-wave–slow-wave problems. *Mon. Weather Rev.* **140**: 1307–1325.
- Gal-Chen T, Somerville R. 1975. On the use of a coordinate transformation for the solution of the Navier–Stokes equations. *J. Comput. Phys.* **17**: 209–228.
- Guo W, Nair R, Qiu J-M. 2014. A conservative semi-Lagrangian discontinuous Galerkin scheme on the cubed sphere. *Mon. Weather Rev.* **142**: 457–475.
- Iske A, Kaser M. 2004. Conservative semi-Lagrangian advection on adaptive unstructured meshes. *Numer. Methods Partial Differ. Equ.* **20**: 388–411.
- Katta K, Nair R, Kumar V. 2015. High-order finite-volume transport on the cubed sphere: Comparison between 1D and 2D reconstruction schemes. *Mon. Weather Rev.* **143**: 2937–2954.
- Kent J, Ullrich P, Jablonowski C. 2014. Dynamical core model intercomparison project: Tracer transport test cases. *Q. J. R. Meteorol. Soc.* **140**: 1279–1293.
- Lashley R. 2002. 'Automatic generation of accurate advection schemes on unstructured grids and their application to meteorological problems', PhD thesis. Departments of Mathematics and Meteorology, University of Reading: UK.
- Lauritzen PH, Nair R, Ullrich P. 2010. A conservative semi-Lagrangian multi-tracer transport scheme (CSLAM) on the cubed-sphere grid. *J. Comput. Phys.* **229**: 1401–1424.
- Lauritzen PH, Skamarock WC, Prather MJ, Taylor MA. 2012. A standard test case suite for two-dimensional linear transport on the sphere. *Geosci. Model Dev.* **5**: 887–901.
- Lauritzen PH, Ullrich PA, Jablonowski C, Bosler PA, Calhoun D, Conley AJ, Enomoto T, Dong L, Dubey S, Guba O, Hansen AB, Kaas E, Kent J, Lamarque J-F, Prather MJ, Reinert D, Shashkin VV, Skamarock WC, Sørensen B, Taylor MA, Tolstykh MA. 2014. A standard test case suite for two-dimensional linear transport on the sphere: Results from a collection of state-of-the-art schemes. *Geosci. Model Dev.* **7**: 105–145.
- Leonard B, Lock A, MacVean M. 1995. The NIRVANA scheme applied to one-dimensional advection. *Int. J. Numer. Methods Heat Fluid Flow* **5**: 341–377.
- Leonard B, Lock A, MacVean M. 1996. Conservative explicit unrestricted-time-step multidimensional constancy-preserving advection schemes. *Mon. Weather Rev.* **124**: 2585–2606.
- Lin S-J. 2004. A 'vertically Lagrangian' finite-volume dynamical core for global models. *Mon. Weather Rev.* **132**: 2293–2307.
- Lin S, Rood R. 1996. Multidimensional flux-form semi-Lagrangian transport schemes. *Mon. Weather Rev.* **124**: 2046–2070.
- Lipscomb W, Ringler T. 2005. An incremental remapping transport scheme on a spherical geodesic grid. *Mon. Weather Rev.* **133**: 2335–2350.
- Lock S, Wood N, Weller H. 2014. Numerical analyses of Runge–Kutta implicit-explicit schemes for horizontally explicit, vertically implicit solutions of atmospheric models. *Q. J. R. Meteorol. Soc.* **140**: 1654–1669.
- Melvin T, Dubal M, Wood N, Staniforth A, Zerroukat M. 2010. An inherently mass-conserving semi-implicit semi-Lagrangian discretisation of the non-hydrostatic vertical slice equations. *Q. J. R. Meteorol. Soc.* **137**: 799–814.
- Miura H. 2007. An upwind-biased conservative advection scheme for spherical hexagonal-pentagonal grids. *Mon. Weather Rev.* **135**: 4038–4044.
- Nair RD, Lauritzen PH. 2010. A class of deformational flow test cases for linear transport problems on the sphere. *J. Comput. Phys.* **229**: 8868–8887.
- OpenFOAM. 2016. 'The open source CFD toolbox'. The OpenCFD Foundation. <https://www.openfoam.org> (accessed 26 July 2017).
- Pudykiewicz J, Staniforth A. 1984. Some properties and comparative performance of the semi-Lagrangian method of Robert in the solution of the advection-diffusion equation. *Atmos. Ocean* **22**: 283–308.
- Purnell D. 1976. Solution of the advective equation by upstream interpolation with a cubic spline. *Mon. Weather Rev.* **104**: 42–48.
- Putman W, Lin S-J. 2007. Finite-volume transport on various cubed-sphere grids. *J. Comput. Phys.* **227**: 55–78.
- Schär C, Leuenberger D, Fuhrer O, Lüthi D, Girard C. 2002. A new terrain-following vertical coordinate formulation for atmospheric prediction models. *Mon. Weather Rev.* **130**: 2459–2480.
- Shaw J, Weller H, Methven J, Davies T. 2017. Multidimensional method-of-lines transport for atmospheric flows over steep terrain using arbitrary meshes. *J. Comput. Phys.* **344**: 86–107. <https://dx.doi.org/10.1016/j.jcp.2017.04.061>.
- Skamarock WC, Gassmann A. 2011. Conservative transport schemes for spherical geodesic grids: High-order flux operators for ODE-based time integration. *Mon. Weather Rev.* **139**: 2962–2975.
- Staniforth A, Thuburn J. 2012. Horizontal grids for global weather and climate prediction models: A review. *Q. J. R. Meteorol. Soc.* **138**: 1–26.
- Thuburn J, Cotter C, Dubos T. 2014. A mimetic, semi-implicit, forward-in-time, finite-volume shallow-water model: Comparison of hexagonal-icosahedral and cubed sphere grids. *Geosci. Model Dev.* **7**: 909–929.
- Ullrich P, Norman M. 2014. The flux-form semi-Lagrangian spectral element (FF-SLSE) method for tracer transport. *Q. J. R. Meteorol. Soc.* **140**: 1069–1085.
- Wan H, Giorgetta M, Zängl G, Restelli M, Majewski D, Bonaventura L, Fröhlich K, Reinert D, Rípodas P, Kornbluh L, Förstner J. 2013. The ICON-1.2 hydrostatic atmospheric dynamical core on triangular grids—Part 1: Formulation and performance of the baseline version. *Geosci. Model Dev.* **6**: 735–763.
- Weller H, Shahrokh A. 2014. Curl-free pressure gradients over orography in a solution of the fully compressible Euler equations with implicit treatment of acoustic and gravity waves. *Mon. Weather Rev.* **142**: 4439–4457.
- Weller H, Weller HG, Fournier A. 2009. Voronoi, Delaunay and block structured mesh refinement for solution of the shallow-water equations on the sphere. *Mon. Weather Rev.* **137**: 4208–4224.
- Weller H, Thuburn J, Cotter CJ. 2012. Computational modes and grid imprinting on five quasi-uniform grids of the sphere. *Mon. Weather Rev.* **140**: 2734–2755.
- Zerroukat M, Wood N, Staniforth A. 2004. SLICE-S: A semi-Lagrangian inherently conserving and efficient scheme for transport problems on the sphere. *Q. J. R. Meteorol. Soc.* **130**: 2649–2664.